The Mixed Vertex Packing Problem [1]

Alper Atamtürk
George L. Nemhauser
Martin W.P. Savelsbergh

School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0205

Abstract

We study the polyhedral structure of a new model for various bottleneck resource allocation problems in which the resource consumption is determined by the activity with the maximum usage. The model, called the mixed vertex packing problem, is a generalization of the vertex packing problem having both binary and bounded continuous variables. We give valid inequalities and separation algorithms for them. We present computational results that show the effectiveness of the valid inequalities in solving mixed vertex packing and general mixed 0-1 integer problems.

KEY WORDS: conflict graphs, vertex packing, valid inequalities, mixed integer programming.

April 1998

1 Introduction

We present a new model for various bottleneck resource allocation problems in which the resource consumption is determined by the activity with the maximum usage. We provide examples arising in the telecommunications industry, in equipment leasing, and in solving general mixed 0-1 integer problems. The model, called the *mixed vertex packing problem* (MVPP), is a generalization of the vertex packing problem having both binary and bounded continuous variables. Formally, MVPP is given by

$$\max\{cx + dy : \ (x,y) \in \text{MVP}\},$$

where

$$\begin{aligned} \text{MVP} = \{\ x \in \mathbb{B}^n, y \in \mathbb{R}^m : x_i + x_j &\le 1, & (i,j) &\in E \\ a_{ik}x_i + y_k &\le u_k, & (i,k) &\in F \\ 0 \le y_k &\le u_k, & k &\in M\}. \end{aligned}$$

We use N to denote the index set of binary variables with $n = |N|$, and M to denote the index set of continuous variables with $m = |M|$. Inequalities over $E \subseteq \{(i,j) : i,j \in N\}$ are called *binary edge inequalities*, whereas the inequalities over $F \subseteq \{(i,k) : i \in N, k \in M\}$ are called *mixed*

[1] This research is supported, in part, by NSF Grant DMI-9700285 to the Georgia Institute of Technology.

edge inequalities. We assume $u_k < \infty$, for all $k \in M$. In order to eliminate uninteresting cases, we will also assume that $u_k > 0$, otherwise $y_k = 0$. Furthermore $0 < a_{ik} \leq u_k$, otherwise either $a_{ik}x_i + y_k \leq u_k$ is redundant or x_i is zero in every feasible solution. Without loss of generality, we assume that $c_i > 0$ for all $i \in N$, and $d_k > 0$ for all $k \in M$, otherwise there is an optimal solution with $x_i = 0$ if $c_i \leq 0$ and $y_k = 0$ if $d_k \leq 0$. Observe that an arbitrary inequality $ax_i + by_k \leq c$ with nonnegative data can be put into the form $a_{ik}x_i + y_k \leq u_k$, by writing it as $(u_k - \frac{c-a}{b})x_i + y_k \leq u_k$ after reducing u_k to c/b, if $u_k > c/b$. Similarly, $ax_i + bx_j \leq c$ can be put into the form $x_i + x_j \leq 1$ if $a + b > c$, otherwise it is redundant. Also note that a simple variable upper bound constraint $y_i \leq u_i x_i$ can be viewed as a special case of the mixed edge inequalities by complementing the binary variable x_i.

Since there are two variables in each constraint, MVP can be represented by a graph $G = (N \cup M, E \cup F)$ and weights on F to denote the conflicts and on M to denote the upper bounds. G is called a *mixed conflict graph* because the edges and the weights represent the conflicts between pairs of variables. In a mixed conflict graph, there are two types of vertices: binary vertices for binary variables and continuous vertices for continuous variables.

Although the NP-hard vertex packing problem is one of the most studied problems in combinatorial optimization ([5, 7, 10, 11, 12, 14] to mention a few), the mixed vertex packing problem apparently has not been studied in the literature. Johnson [8] shows how to strengthen simple variable upper bound constraints in the presence of binary edges and gives a special case of the mixed clique inequalities we derive in the sequel. Our treatment here is more general and extensive.

MVPP can alternatively be written as

$$\max_{P} \sum_{i \in P} c_i + \sum_{k \in M} d_k(u_k - \max_{i \in P} a_{ik}), \tag{1}$$

where P is a packing in $G(N)$, the subgraph induced by N. Since the term $\sum_{k \in M} d_k u_k$ in (1) is a constant, MVPP can be used to model various bottleneck resource allocation problems, in which the resource consumption is determined by an activity with maximum usage.

A direct application of MVPP comes from the telecommunications industry. Consider a communications company that has a set of transmitters M, each used to provide a different type of service to a set of possible customers N. The amount of resource used by a transmitter is determined by the farthest customer that the transmitter serves, since all the customers closer to the transmitter share the resource used to provide service to this customer. The company is faced with the problem of deciding which customers to serve so as to maximize the sum of profits from the customers minus the cost of resources used, which is a linear function of the broadcast diameter. In Figure 1, the circles denote the customers and the squares denote the transmitters.

Another application arises in equipment leasing. Planning to lease equipment to be used for a set of alternative projects requires balancing the tradeoff between the leasing cost and the profits from the projects. Clearly, the capacity of the equipment should satisfy the maximum requirements by the projects during the leasing horizon. Timing and other constraints may lead to pairwise conflicts between the projects. Assuming a linear cost for acquiring capacity, this problem can also be modeled as a mixed vertex packing problem.

Yet another motivation for studying this problem is that one can derive a mixed vertex packing relaxation for a general mixed 0-1 integer problem (MIP), just as one can derive a vertex packing relaxation for a pure 0-1 integer problem, see for example [2, 4]. Therefore, valid inequalities for

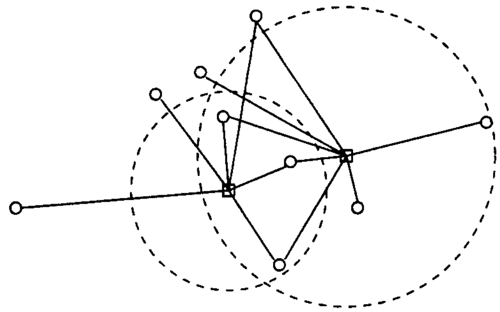

Figure 1: Telecommunications problem.

MVP are also valid for the original problem. We present a small example illustrating the derivation of the MVP relaxation.

Example. Consider the mixed 0-1 integer set

$$
\mathrm{S} = \{\, x \in \mathbb{B}^4, y \in \mathbb{R}^3_+ : \begin{array}{rrrrrrrl}
3x_1 & & & +6x_4 & +y_1 & & & \leq 9 \\
 & & 13x_3 & & -2y_1 & +2y_2 & +3y_3 & \leq 6 \\
2x_1 & +5x_2 & +3x_3 & & & & & \leq 6 \\
 & & & & \multicolumn{4}{r}{y_1 \leq 9,\; y_2 \leq 10,\; y_3 \leq 8\ \}.}
\end{array}
$$

The following logical implications, which can be found by probing [13], are valid for S:

$$
\begin{array}{ll}
x_1 = 1 \Rightarrow & x_2 = 0, y_1 \leq 6 \Rightarrow \ y_2 \leq 9,\ y_3 \leq 6, \\
x_2 = 1 \Rightarrow & x_1 = 0,\ x_3 = 0, \\
x_3 = 1 \Rightarrow & x_2 = 0, y_1 \geq \frac{7}{2} \rightarrow x_4 = 0, \\
x_4 = 1 \Rightarrow & y_1 \leq 3 \Rightarrow x_3 = 0,\ y_2 \leq 6,\ y_3 \leq 4.
\end{array}
$$

Writing these implications as linear inequalities gives us a packing relaxation of S:

$$
\mathrm{MVP} = \{\, x \in \mathbb{B}^4, y \in \mathbb{R}^3_+ : \begin{array}{rrrrrrrll}
3x_1 & & & & +y_1 & & & \leq & 9 \\
 & & & 6x_4 & +y_1 & & & \leq & 9 \\
1x_1 & & & & & +y_2 & & \leq & 10 \\
 & & & 4x_4 & & +y_2 & & \leq & 10 \\
2x_1 & & & & & & +y_3 & \leq & 8 \\
 & & & 4x_4 & & & +y_3 & \leq & 8 \\
x_1 & +x_2 & & & & & & \leq & 1 \\
 & x_2 & +x_3 & & & & & \leq & 1 \\
 & & x_3 & +x_4 & & & & \leq & 1\ \}.
\end{array}
$$

Figure 2 shows the mixed conflict graph for the packing relaxation of S in our example. We use circles to denote the binary vertices and squares for the continuous vertices. Note that there are no edges between continuous vertices.

This paper is organized as follows. In Section 2, we study an important special case in which the binary vertices form an independent set. We show that, in this case, the mixed vertex packing problem can be polynomially reduced to a vertex packing problem on a comparability graph and MVPP is solvable in polynomial time. In Section 3, we study the polyhedral structure of the mixed vertex packing polytope. We derive several classes of valid inequalities for this polytope and

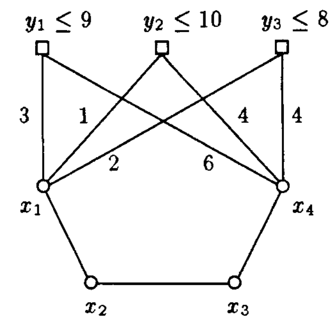

Figure 2: Mixed conflict graph of S.

give separation algorithms for them. Furthermore, we show how to strengthen these inequalities through coefficient improvement and lifting. In Section 4, we present computational experiments that indicate the effectiveness of the new valid inequalities in solving the MVPPs and general 0-1 MIPs. Finally, in Section 5, we conclude with some extensions currently under investigation.

The following notation will be used in the remainder of the paper. For $i \in N \cup M$

$$N(i) = \{j \in N : (i,j) \in E \cup F\}, \qquad M(i) = \{k \in M : (i,k) \in F\}.$$

Thus for vertex i, $N(i)$ denotes the set of binary vertices adjacent to i, whereas $M(i)$ denotes the set of continuous vertices adjacent to i.

2 The case of independent binary vertices

Since the vertex packing problem is a special case of MVPP, MVPP is NP-hard, in general. However, in this section, we will show that when the binary vertices form an independent set, i.e. $E = \emptyset$, MVPP is polynomially solvable. Recall that this is the situation for the telecommunications example given in Section 1. Since $E = \emptyset$, the conflict graph is bipartite. We show that, in this case, MVPP is polynomially solvable by reducing MVPP to a vertex packing problem on a comparability graph.

The possible values of y_i, $i \in M$, in an optimal extreme point of *conv*(MVP) are u_i and $u_i - a_{ij}$, $j \in N(i)$. For simplicity of notation, we assume that for $i \in M$ all a_{ij} are distinct. We construct a graph $G' = (\{V, V_1, V_2, \dots, V_m\}, E')$, where $v_j \in V$ if and only if $j \in N$; $v_{ik} \in V_i$ if and only if $\bar{y}_{ik}$ is a value y_i can take on in an optimal extreme point; and $(v_j, v_{ik}) \in E'$ if and only if $a_{ij} + \bar{y}_{ik} > u_i$. We index v_{ik}, $k = 0, 1, \dots, |N(i)|$ in nondecreasing order of $\bar{y}_{ik}$. So, $\bar{y}_{i0} = u_i - \max_{j \in N(i)} a_{ij}$ and $\bar{y}_{i|N(i)|} = u_i$. Furthermore, $(v_{ik}, v_{il}) \in E'$ for all i, k, l, and $(v_i, v_j) \in E'$ if and only if $(i,j) \in E$. In other words, we represent all possible values y_i can take on in an optimal basic feasible solution with a clique on V_i, and we have an edge between a vertex v_j in V and a vertex v_{ik} representing a value $\bar{y}_{ik}$ for y_i if and only if y_i cannot have value $\bar{y}_{ik}$ when x_j is one. In Figure 3, we give a small example to illustrate the reduction. Let $w' : V(G') \mapsto \mathbb{R}_+$ be a weight function defined as $w'(v_{ik}) = d_i \bar{y}_{ik}$, $v_{ik} \in V_i$ and $w'(v_j) = c_j$, $v_j \in V$. Due to the construction of G' and the weight function, for every optimal mixed vertex packing on G there is a corresponding optimal vertex packing on G' with the same value, and vice versa. Observe that the reduction is given for the general MVPP.

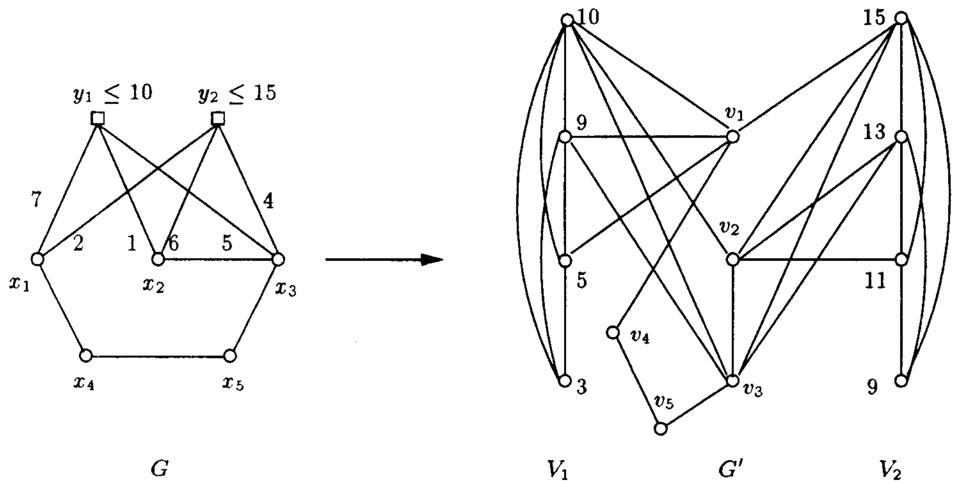

Figure 3: Reduction to vertex packing problem.

A graph is a *comparability graph* if its edges can be oriented to obtain a transitive acyclic directed graph $D = (V, A)$; that is, an acyclic directed graph with the property that if $(u, v) \in A$ and $(v, w) \in A$ then $(u, w) \in A$.

Theorem 2.1 *If the binary vertices of G are independent, then G' is a comparability graph.*

Proof. G is a comparability graph with edges oriented from N to M. We use the same orientation for G', i.e. we orient the edges between v_i and V_j towards V_j. Then we orient the edges in clique $G(V_j)$ 'upwards', thus (v_{jk}, v_{jl}) is a directed arc if and only if $\bar{y}_{jk} < \bar{y}_{jl}$.

With this orientation, G' is a comparability graph. Observe that all edges in the cutset $\delta(V_j)$, $j \in M$ are directed towards V_j. Now, suppose $(a, b), (b, c) \in E'$. Since b has an incoming arc and an outgoing arc, $b \notin V$. So $b \in V_j$, for some $j \in M$. If $(a, b) = (v_i, v_{jk})$, $(v_i, v_{jl}) \in E'$ for all $l > k$ and therefore $(a, c) \in E'$; else $(a, b) = (v_{jk}, v_{jl})$ but since V_i is a clique, $(a, c) \in E'$. □

Corollary 2.2 *The mixed vertex packing problem can be solved in polynomial time if the binary vertices are independent.*

Proof. The weighted vertex packing problem is solvable in polynomial time on comparability graphs [6]. □

Note that in the proof of Theorem 2.1, we only used the bipartiteness of G. Therefore Theorem 2.1 extends immediately to general bipartite graphs. Hence, we have the following result.

Theorem 2.3 *The mixed vertex packing problem can be solved in polynomial time on bipartite graphs.*

3 MVP polytope

In this section, we study the facial structure of the mixed vertex packing polytope, *conv*(MVP), and derive valid inequalities for it. Let LMVP be the linear relaxation of MVP. Thus,

$$
\begin{aligned}
\text{LMVP} = \{ \ (x, y) \in \mathbb{R}^{n+m} : & \\
x_i + x_j \leq 1, \quad & (i,j) \in E && (2)\\
a_{ik}x_i + y_k \leq u_k, \quad & (i,k) \in F && (3)\\
0 \leq x_i \leq 1, \quad & i \in N && (4)\\
0 \leq y_k \leq u_k, \quad & k \in M\}. && (5)
\end{aligned}
$$

Below we summarize some basic results.

Proposition 3.1

1. *Conv(MVP) is full-dimensional.*
2. $x_i \geq 0,\ i \in N$ *and* $y_k \geq 0,\ k \in M$ *are facet-defining for conv(MVP).*
3. $x_i \leq 1,\ i \in N$ *defines a facet of conv(MVP) if and only if* $N(i) = \emptyset$ *and* $a_{ik} < u_k$ *for all* $k \in M(i)$.
4. $y_k \leq u_k,\ k \in M$ *defines a facet of conv(MVP) if and only if* $M(k) = \emptyset$.
5. $x_i + x_j \leq 1$ *defines a facet of conv(MVP) if and only if* $N(i) \cap N(j) = \emptyset$ *and* $\min\{a_{ik}, a_{jk}\} < u_k$ *for all* $k \in M(i) \cup M(j)$.
6. $a_{ik}x_i + y_k \leq u_k$ *defines a facet of conv(MVP) if and only if* $N(i) \cap N(k) = \emptyset$ *and* $a_{ik} = \max_{j \in N(k)} a_{jk}$.

In the following theorem, we characterize the graphs for which the linear relaxation LMVP is sufficient to describe *conv*(MVP).

Theorem 3.2 *Inequalities (2), (3), (4), and (5) are sufficient to describe conv(MVP) if and only if G is bipartite and* $a_{ik} = a_k$, *for all* $i \in N(k)$, *for all* $k \in M$.

Proof. [$\Rightarrow$] Suppose $a_{ik} < a_{jk}$ for some $k \in M$. In Proposition 3.4 we show that $(a_{jk} - a_{ik})x_j + a_{ik}x_i + y_k \leq u_k$ is valid for *conv*(MVP) and it dominates $a_{ik}x_i + y_k \leq u_k$. Now, suppose $a_{ik} = a_k$, for all $i \in N(k)$ for all $k \in M$ but G is not bipartite. In that case, consider the odd cycle given in Figure 4. It is easily seen that $(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, u - \frac{a}{2})$ is a fractional basic feasible solution of LMVP if u is the upper bound of the continuous variable.
[$\Leftarrow$] Define $y'_k = y_k/a_k$, and rewrite (3) as $x_i + y'_k \leq u_k/a_k$. Since G is bipartite, the $\{0,1\}$ constraint matrix of (2), (3), (4), and (5) is totally unimodular. Let B be a full-column-rank submatrix given by the inequalities that define a basic feasible solution. Let b be the right hand side of these inequalities and define b_I to be the integral vector obtained by rounding down the elements of b and $b_F = b - b_I$. Since B^{-1} is integral, so is $B^{-1}b_I$. To see that $B^{-1}b_F$ has entries equal to zero for all the x variables, observe that the edge inequalities of B represent a forest. Otherwise there

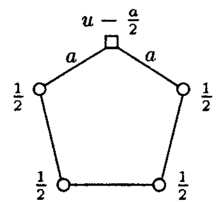

Figure 4: Fractional basic feasible solution.

is an even cycle and B does not have full column rank. Then, in each connected component, there is a variable with a tight bound since the number of edge inequalities is one less than the number of variables. If some x_i has a tight bound (either lower or upper) or if some y'_k is tight at its upper bound then $B^{-1}b_F$ has 0 entries for all x_i in that component. Variable y'_k can be tight at its lower bound only if k is an isolated vertex or if $a_k = u_k$, in which case $B^{-1}b_F$ has 0 entries for all x_i in that component as well. Note that if $a_k < u_k$ and $x_i + y'_k \leq u_k/a_k$ is a tight edge inequality of a component, then $y'_k \neq 0$ in a feasible solution since $y'_k = 0$ would imply $x_j = u_k/a_k > 1$ for all $j \in N(i)$. Therefore, $B^{-1}b_I + B^{-1}b_F$ is integral for all the x variables. □

3.1 Valid inequalities

There is a natural vertex packing relaxation of MVP, defined on the subgraph induced by the binary vertices. Valid inequalities for this vertex packing relaxation are valid for MVP as well.

Proposition 3.3 *Let MVP(N) denote the projection of MVP onto the space of binary variables. If*

$$\sum_{i \in S} \alpha_i x_i \leq r \tag{6}$$

for some $S \subseteq N$ is a valid inequality for MVP(N), then it is valid for MVP as well. If (6) is facet-defining for MVP(N), then it is also facet-defining for MVP if for all $k \in M$, there exists a packing $P_k \subseteq S$ satisfying (6) at equality with $a_{ik} < u_k$ for all $i \in P_k \cap N(k)$.

Proof. The inequality is valid for MVP since $a_{ik} > 0$ for all i and k. If (6) is facet-defining for MVP(N), there exists n affinely independent points in MVP(N) satisfying (6) at equality. Let e_i be the i^{th} unit vector. These n points together with $\sum_{i \in P_k} e_i + (u_k - \max_{i \in P_k} a_{ik})e_k$, for $k \in M$, make up $n + m$ affinely independent points in $\{(x, y) \in MVP : \sum_{i \in S} b_i x_i = r\}$. □

For a vertex k, a subgraph consisting of vertices k and $T \subseteq N(k)$ and the edges between k and T, is said to be a *star* of vertex k. Now, we give the first class of new valid inequalities for MVP.

Proposition 3.4 *For $k \in M$, let $T = \{i_1, i_2, \ldots, i_t\}$ be a subset of $N(k)$ such that $a_{i_{j-1}k} < a_{i_j k}$ for $j = 2, 3, \ldots, t$. Then the star inequality*

$$\sum_{i \in T} \bar{a}_{ik} x_i + y_k \leq u_k \tag{7}$$

with

$$\begin{aligned}\bar{a}_{i_1k} &= a_{i_1k},\\ \bar{a}_{i_jk} &= a_{i_jk} - a_{i_{j-1}k},\ j = 2,3,\dots,t,\end{aligned}$$

is valid for conv(MVP).

Proof. Let $(\bar{x},\bar{y}) \in$ MVP, $S = \{i \in T : \bar{x}_i = 1\}$ and $j^* = \max_{1\le j\le t}\{j : i_j \in S\}$. Then

$$\sum_{i\in T} \bar{a}_{ik}\bar{x}_i + \bar{y}_k \le \sum_{i\in S} \bar{a}_{ik} + (u_k - a_{i_{j^*}k}) \le a_{i_{j^*}k} + (u_k - a_{i_{j^*}k}) = u_k.\ QED$$

Theorem 3.5 *The star inequality (7) is facet-defining for conv(MVP) if* $a_{i_tk} = \max_{j\in N(k)} a_{jk}$, *and* $N(i) = \emptyset$ *for all* $i \in T$.

Proof. Suppose $N(k) = \{1, 2, \dots, l\}$ is indexed so that $a_{1k} \le a_{2k} \le \dots \le a_{lk}$. In order to show that (7) is facet-defining, we will give $n + m$ linearly independent points in $\{(x,y) \in MVP : \sum_{i\in T} \bar{a}_{ik}x_i + y_k = u_k\}$. Consider

$$\begin{aligned}p_k &= u_k e_k,\\ p_i &= u_k e_k + u_i e_i,\ i \in M \setminus \{k\},\\ q_i &= u_k e_k + e_i,\ i \in N \setminus N(k),\\ w_i &= \sum_{j\in N(k):j\le i} e_j + (u_k - a_{ik})e_k, i \in T,\\ z_i &= \sum_{j\in N(k):j\le j(i),j\ne i} e_j + (u_k - a_{j(i)k})e_k,\ i \in N(k)\setminus T,\end{aligned}$$

where for $i \in N(k) \setminus T$, $j(i) = \min_{1\le j\le t}\{i_j \in T : a_{ik} \le a_{i_jk}\}$. Note that $j(i)$ is well defined since $a_{i_tk} = \max_{j\in N(k)} a_{jk}$. Suppose these points satisfy

$$\sum_{i\in N} \beta_i x_i + \sum_{i\in M} \pi_i y_i = \pi_o. \tag{8}$$

By comparing p_k with p_i, we see that $\pi_i = 0,\ i \in M \setminus \{k\}$. Similarly, $\beta_i = 0,\ i \in N \setminus N(k)$. Furthermore, by comparing z_i with $w_{j(i)}$, we see that $\beta_i = 0,\ i \in N(k)\setminus T$. Then, (8) is of the form

$$\sum_{i\in T} \beta_i x_i + \pi_k y_k = \pi_o. \tag{9}$$

From p_k, we have $\pi_k u_k = \pi_o$ and from w_{i_1}, we have $\beta_{i_1} + \pi_k(u_k - a_{i_1k}) = \pi_o = \pi_k u_k$ or $\beta_{i_1} = a_{i_1k}\pi_k$. Now, suppose $\beta_{i_h} = (a_{i_hk} - a_{i_{h-1}k})\pi_k$ for all $h \le j-1$, then from w_{i_j}, we have $\pi_k a_{i_{j-1}} + \beta_{i_j} + \pi_k(u_k - a_{i_jk}) = \pi_k u_k$ or $\beta_{i_j} = (a_{i_jk} - a_{i_{j-1}k})\pi_k$. Then, (9) is of the form

$$\sum_{j=1}^{t} \pi_k(a_{i_jk} - a_{i_{j-1}k})x_{i_j} + \pi_k y_k = \pi_k u_k$$

with a_{i_0k} defined to be 0. This proves that the points given above define (7) up to a scalar, hence they are linearly independent. Furthermore, since the origin is a feasible point of MVP and $u_k > 0$, (7) induces a proper face of *conv*(MVP). □

Observe that the mixed edge inequalities (3) are dominated by the star inequalities (7). In Section 2, we showed that MVPP is solvable in polynomial time if the binary vertices are independent. In this case, the star inequalities together with the upper and lower bound inequalities give the convex hull of MVP.

Theorem 3.6 *If $E = \emptyset$, inequalities (4), (5), and (7) are sufficient to describe conv(MVP).*

Proof. Given an arbitrary objective function $(c, d) \neq (0,0)$, let $(\bar{x}^l, \bar{y}^l)$, $l \in \mathcal{O}$ be the optimal solutions to MVPP. We will prove the theorem by showing that there exists an inequality among (4), (5), and (7) that is satisfied at equality for *all* $l \in \mathcal{O}$. If $c_j < 0$ for some $j \in N$ then $\bar{x}^l_j = 0$ for all $l \in \mathcal{O}$. Therefore, in the following we assume $c_j \geq 0$. Similarly, if $d_k < 0$ for some $k \in M$ then $\bar{y}^l_k = 0$ for all $l \in \mathcal{O}$. Furthermore, we can remove $k \in M$ such that $d_k = 0$ and $j \in N$ such that $c_j = 0$ from the graph and restrict our analysis to the remaining subgraph. In this subgraph, if there is an isolated vertex, the result follows, since the upper bound constraint is satisfied at equality for all of the optimal solutions for that vertex. So, for the rest of the discussion, we can assume $c_j, d_k > 0$ and $N(i) \neq \emptyset$ for all $i \in N \cup M$. To simplify the argument, we will also assume that for $k \in M$, $a_{jk}, j \in N(k)$ are distinct.

We define $S^l = \{j \in N : \bar{x}^l_j = 1\}$, $l \in \mathcal{O}$ and for an arbitrary $t \in M$, let $S^l_t = S^l \cap N(t)$. If $S^l_t = \emptyset$ for all $l \in \mathcal{O}$, we are done since $\bar{y}^l_t = u_t$ for all $l \in \mathcal{O}$. Otherwise, let $T = \{j \in N(t) : j = \text{argmax}_{k \in S^l_t} a_{kt}, \text{ for } l \in \mathcal{O}\}$. Furthermore, we define $T' = T \cup \text{argmax}_{k \in N(t)} a_{kt}$. We claim that the star inequality

$$\sum_{k \in T'} \bar{a}_{kt} x_k + y_t \leq u_t \tag{10}$$

is satisfied at equality for all $l \in \mathcal{O}$. Consider some $p \in \mathcal{O}$ and let $j = \text{argmax}_{k \in S^p_t} a_{kt}$. Inequality (10) is satisfied at equality by $(\bar{x}^p, \bar{y}^p)$ if and only if

$$\sum_{k \in T' : a_{kt} < a_{jt}} \bar{a}_{kt} = a_{jt}$$

since $d_t > 0$ and $\bar{y}^p_t = u_t - a_{jt}$. Therefore, (10) has positive slack for $(\bar{x}^p, \bar{y}^p)$ if and only if there exists $i \in T$ such that $a_{it} < a_{jt}$ and $i \notin S^p_t$. Suppose there is such an $i \in T$. By definition of T there is an optimal solution $(\bar{x}^q, \bar{y}^q)$ such that $i = \text{argmax}_{k \in S^q_t} a_{kt}$.

In order to arrive at a contradiction, we construct a solution with a larger objective value then of $(\bar{x}^p, \bar{y}^p)$. Let $z(S) = \sum_{k \in S} c_k - \sum_{r \in M} d_r \max_{k \in S} a_{kr}$ for $S \subseteq N$. We show that $z(S^p \cup S^q) > z(S^p)$. Let $K = S^p \cap S^q$, $M^q = \{r \in M : \max_{k \in S^q} a_{kr} > \max_{k \in S^p} a_{kr}\}$, $M^p = M \setminus M^q$. Then, we have

$$\begin{aligned}
z(S^p \cup S^q) &= \sum_{k \in S^p} c_k - \sum_{r \in M^p} d_r \max_{k \in S^p} a_{kr} + \sum_{k \in S^q \setminus K} c_k - \sum_{r \in M^q} d_r \max_{k \in S^q \setminus K} a_{kr} \\
&= z(S^p) + \sum_{k \in S^q \setminus K} c_k - \sum_{r \in M^q} d_r (\max_{k \in S^q \setminus K} a_{kr} - \max_{k \in S^p} a_{kr}) \\
&\geq z(S^p) + \sum_{k \in S^q \setminus K} c_k - \sum_{r \in M^q} d_r (\max_{k \in S^q} a_{kr} - \max_{k \in K} a_{kr}).
\end{aligned}$$

However,

$$\begin{aligned}\sum_{k\in S^q\setminus K} c_k - \sum_{r\in M^q} d_r(\max_{k\in S^q} a_{kr} - \max_{k\in K} a_{kr}) &> \sum_{k\in S^q\setminus K} c_k - \sum_{r\in M} d_r(\max_{k\in S^q} a_{kr} - \max_{k\in K} a_{kr}) \\ &= z(S^q) - z(K) \\ &\geq 0.\end{aligned}$$

The strict inequality holds because $\max_{k\in S^q} a_{kr} \geq \max_{k\in K} a_{kr}$ for all $r \in M$ ($K \subset S^q$), and $d_t(\max_{k\in S^q} a_{kt} - \max_{k\in K} a_{kt}) > 0$. Note that $t \in M^p$ since, by assumption, $a_{it} < a_{jt}$. Also, since S^q is optimal, $z(S^q) \geq z(K)$. Therefore it must be the case that $z(S^p \cup S^q) > S^p$, which contradicts the optimality of S^p. □

The next two classes of inequalities are generalizations of the clique and odd cycle inequalities [11, 12] for the vertex packing problem, respectively.

Theorem 3.7 *If $K \subseteq N(k)$, $k \in M$ induces a clique then the mixed clique inequality*

$$\sum_{i\in K} a_{ik}x_i + y_k \leq u_k \tag{11}$$

is valid for conv(MVP). It is facet-defining for conv(MVP) if and only if for all $j \in N(k)\setminus K$, there is $i \in K \setminus N(j)$ such that $a_{jk} \leq a_{ik}$.

Proof. The validity of (11) is obvious since at most one of the variables in K can have value one.
[⇒] Suppose for some $j \in N(k) \setminus K$, $a_{jk} > a_{ik}$ for $i \in K \setminus N(j)$. Then $\sum_{i\in K} a_{ik}x_i + (a_{jk} - \max_{i\in K\setminus N(j)} a_{ik})x_j + y_k \leq u_k$ is valid and dominates (11).
[⇐] Let $i(j) = \text{argmax}_{i\in K\setminus N(j)} a_{ik}$ for $j \in N(k) \setminus K$. Then $e_i + (u_k - a_{ik})e_k$, $i \in K$, $e_{i(j)} + (u - a_{i(j)k})e_k + e_j$, $j \in N(k)\setminus K$, $e_j + u_k e_k$, $j \in N \setminus N(k)$ and $u_k e_k + u_i e_i$, $i \in M \setminus \{k\}$, $u_k e_k$ are $n+m$ linearly independent points of $\{(x,y) \in \text{MVP} : \sum_{i\in K} a_{ik}x_i + y_k = u_k\}$. □

Theorem 3.8 *Let $C \subseteq E \cup F$ be the set of edges of an odd cycle in G, C_B be the set of binary vertices on the cycle, and C_C the set of continuous vertices on the cycle. The mixed odd cycle inequality*

$$\sum_{j\in C_B}(1 + \sum_{k\in M_j} \frac{a_k^2 - a_k^1}{a_k^1})x_j + \sum_{k\in C_C} y_k/a_k^1 \leq \lfloor \frac{1}{2}(|C_B| - |C_C|)\rfloor + \sum_{k\in C_C} u_k/a_k^1 \tag{12}$$

is valid for conv(MVP), where a_k^1, a_k^2 are the weights of the incident edges to $k \in C_C$ in C, with $a_k^1 \leq a_k^2$ and $M_j = \{k \in M(j) \cap C_C : a_k^2 = a_{jk}\}$.

Proof. We claim that the left hand side value of (12) is maximized by maximizing the number binary variables with value one on the cycle. To see this, consider $k \in C_C$ and its adjacent vertices on the cycle, k_1 and k_2 with edge weights a_k^1, a_k^2, respectively. We analyze the maximum value $\mathcal{L}$ of $y_k/a_k^1 + x_{k_1} + (1 + \frac{a_k^2 - a_k^1}{a_k^1})x_{k_2}$. There are four cases:
Case 1: $(x_{k_1} = 0,\ x_{k_2} = 0)$ $\mathcal{L} = \frac{u_k}{a_k^1}$
Case 2: $(x_{k_1} = 1,\ x_{k_2} = 0)$ $\mathcal{L} = \frac{u_k - a_k^1}{a_k^1} + 1 = \frac{u_k}{a_k^1}$

Case 3: $(x_{k_1} = 0,\ x_{k_2} = 1)$ $\mathcal{L} = \frac{u_k - a_k^2}{a_k^1} + 1 + \frac{a_k^2 - a_k^1}{a_k^1} = \frac{u_k}{a_k^1}$

Case 4: $(x_{k_1} = 1,\ x_{k_2} = 1)$ $\mathcal{L} = \frac{u_k - a_k^2}{a_k^1} + 2 + \frac{a_k^2 - a_k^1}{a_k^1} = \frac{u_k}{a_k^1} + 1.$

Note that $\mathcal{L}$ is constant in all cases except when $x_{k_1} = x_{k_2} = 1$. Therefore, the left hand side of the inequality is maximized by maximizing the number of binary variables at value one on the cycle. Observe that at most $\frac{|C|-1}{2}$ binary variables can take value one on the cycle and we may assume that every continuous vertex is incident to at least one binary vertex having value one. Moreover, for cases 2, 3, and 4, we have the maximum value of $y_k/a_k^1 + \frac{a_k^2 - a_k^1}{a_k^1} x_{k_2} = \frac{u_k}{a_k^1} - 1$. Therefore,

$$\begin{aligned}
\sum_{j \in C_B} (1 + \sum_{k \in M_j} \frac{a_k^2 - a_k^1}{a_k^1}) x_j + \sum_{k \in C_C} y_k/a_k^1 &= \sum_{j \in C_B} x_j + \sum_{k \in C_C} \sum_{j : k \in M_j} (\frac{a_k^2 - a_k^1}{a_k^1}) x_j + \sum_{k \in C_C} y_k/a_k^1 \\
&\leq \frac{1}{2}(|C| - 1) + \sum_{k \in C_C} u_k/a_k^1 - |C_C| \\
&= \frac{1}{2}(|C_B| - |C_C| - 1) + \sum_{k \in C_C} u_k/a_k^1,
\end{aligned}$$

which is equal to the right hand side of (12) since in an odd cycle $|C_B| - |C_C|$ is always positive and odd. □

The odd cycle in Figure 5 has a single binary edge. In this example, $x_1 + \frac{3}{2}x_2 + 2x_3 + \frac{1}{2}y_1 + y_2 \leq \frac{9}{2}$ is a mixed odd cycle inequality.

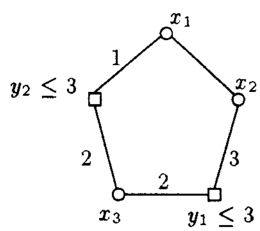

Figure 5: Odd cycle of a mixed conflict graph.

Proposition 3.9 *The mixed odd cycle inequality (12) is facet-defining if G is a chordless odd cycle.*

The proof is similar to the one for the chordless odd cycle inequality for the vertex packing problem and is given in [1].

Example (continued). In our example the valid star inequalities for MVP (hence for S) are

$$\begin{array}{rrrcr}
3x_1 & +3x_4 & +y_1 & \leq & 9 \\
 & 6x_4 & +y_1 & \leq & 9 \\
x_1 & +3x_4 & +y_2 & \leq & 10 \\
 & 4x_4 & +y_2 & \leq & 10 \\
2x_2 & +2x_4 & +y_2 & \leq & 8 \\
 & 4x_4 & +y_2 & \leq & 8
\end{array}$$

and the valid mixed odd cycle inequalities are

$$\begin{array}{cccccccc} x_1 & +x_2 & +x_3 & +2x_4 & +\frac{1}{3}y_1 & \leq & 4 \\ x_1 & +x_2 & +x_3 & +4x_4 & +y_2 & \leq & 11 \\ x_1 & +x_2 & +x_3 & +3x_4 & +\frac{1}{2}y_3 & \leq & 5. \end{array}$$

Although MVP is a relaxation of S, some of the basic feasible solutions of the linear relaxation of S, SL, may not be feasible for the linear relaxation of MVP, MVPL. For example $(\frac{1}{2}, 1, 0, \frac{1}{2}, 7, 10, 0)$ is a basic feasible solution of SL but is not in MVPL. It is cutoff by the edge inequalities $x_1 + x_2 \leq 1$, $x_1 + y_2 \leq 10$, and $4x_4 + y_2 \leq 10$. $(\frac{4}{19}, \frac{15}{19}, \frac{4}{19}, 0, \frac{159}{19}, 10, 0)$ is a basic feasible solution of SL $\cap$ MVPL and is cutoff by $x_1 + x_2 + x_3 + 4x_4 + y_2 \leq 11$ and by $x_1 + 5x_4 + y_2 \leq 10$. □

3.2 Separation

In this section, we discuss the separation problems for the inequalities derived in the previous section. Given a point $(x, y) \in \mathbb{R}^{n+m}$, we are interested in finding a valid inequality violated by this point.

Theorem 3.10 *The separation problem for star inequalities (7) can be solved in polynomial time.*

Proof. For $k \in M$, without loss of generality, suppose $N(k) = \{1, 2, \ldots, l\}$ is indexed so that $a_{1k} \leq a_{2k} \leq \ldots \leq a_{lk}$. We will reduce the separation problem to a longest path problem on an acyclic directed graph with $l + 1$ layers. The graph has a layer for each variable $x_i, i \in N(k)$ and an additional layer "0". The vertices in layer i, $1 \leq i \leq l$, represent all possible sums of coefficients of x_1 to x_i in a star inequality. Layer 0 has a single vertex and represents the "zero" coefficient. Since the sum of all the coefficients in a star inequality is equal to a_{lk}, layer l has a single vertex as well representing coefficient a_{lk}. Arcs from layer $i - 1$ to i represent possible coefficients for variable x_i. From a vertex representing sum s at layer $i - 1$ there are exactly two arcs to layer i. The first one is to the vertex representing sum s at layer i, the second one is to the vertex representing sum a_{ik}. The first arc is for coefficient 0 and the second arc is for coefficient $a_{ik} - s$.

With this construction, if all a_{ik} are distinct, there are $i + 1$ vertices in layer i, $0 \leq i < l$ and a single vertex in layer l, which gives a total of $l(l + 1)/2 + 1$ vertices and l^2 arcs. Furthermore, there are exactly 2^{l-1} different directed paths from layer 0 to layer l, each representing a particular inequality given by (7). If $a_{i-1k} = a_{ik}$, the number of vertices in layers $i - 1$ and i are equal; hence the number of arcs from layer $i - 1$ to layer i is one less than otherwise.

Given $(\bar{x}, \bar{y}) \in \mathbb{R}^{n+m}$, we assign a weight of $c\bar{x}_i$ to an arc representing coefficient c of variable x_i in (7), and a longest path from layer 0 to layer l corresponds to the inequality with the largest left hand side for (7). □

Example. Consider

$$\begin{aligned} S = \{ (x, y) \in \mathbb{B}^4 \times R^1_+ : \; & 1x_1 + y \leq 10, \; 2x_2 + y \leq 10, \\ & 5x_3 + y \leq 10, \; 7x_4 + y \leq 10 \}. \end{aligned}$$

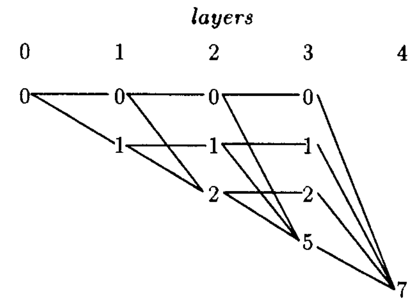

Figure 6: Layered directed graph of S.

We show the layered directed graph of S in Figure 6. Each path from layer 0 to layer 4 represents one of the star inequalities below:

$$\begin{array}{rrrrrl}
x_1 & +x_2 & +3x_3 & +2x_4 & +y & \leq 10 \\
 & 2x_2 & +3x_3 & +2x_4 & +y & \leq 10 \\
x_1 & & +4x_3 & +2x_4 & +y & \leq 10 \\
x_1 & +x_2 & & +5x_4 & +y & \leq 10 \\
x_1 & & & +6x_4 & +y & \leq 10 \\
 & 2x_2 & & +5x_4 & +y & \leq 10 \\
 & & 5x_3 & +2x_4 & +y & \leq 10 \\
 & & & 7x_4 & +y & \leq 10.
\end{array}$$

Exploiting the fact that arcs representing coefficient zero have weight zero, we have the following simple $\Theta(l^2)$ algorithm to separate star inequalities (7).

Algorithm 1 Separation for star inequalities

$\pi_0 \leftarrow 0$
for $j = 1$ **to** l **do**
  $\pi_{a_{jk}} \leftarrow \max_{i:a_{ik}<a_{jk}} \pi_{a_{ik}} + (a_{jk} - a_{ik})\bar{x}_j$
end for
if $\pi_{a_{lk}} + \bar{y}_k > u_k$ **then**
  star inequality, defined by a longest path, is violated
else
  no star inequality is violated
end if

Theorem 3.10 together with Theorem 3.6 provide an alternative proof of polynomial solvability of MVPP for the independent case. Therefore, the separation algorithm leads to an alternative polynomial time algorithm based on linear programming for the problem in this case.

The separation problem for mixed clique inequalities is equivalent to solving a weighted maximum clique problem for each $k \in M$ on the subgraph induced by $N(k)$ and therefore is NP-hard. Given $(\bar{x}, \bar{y}) \in \mathbb{R}^{n+m}$ a most violated mixed clique inequality can be found by solving

$$\max_{k \in M} \left\{ \max_{K \subseteq N(k)} \sum_{j \in K} a_{jk}\bar{x}_j + \bar{y}_k \right\}$$

where $G(K)$ is a clique. It may be feasible to solve this separation problem by enumeration for small graphs since the search for cliques is restricted to adjacent vertices of a single continuous vertex.

Theorem 3.11 *Suppose $a_{jk} = a_k$, for all $j \in N(k)$, for all $k \in M$. Then the separation problem for the mixed odd cycle inequalities (12) can be solved in polynomial time.*

Proof. Consider inequality (12) when weights of all the edges incident to a continuous vertex k are the same, say a_k,

$$\sum_{j \in C_B} x_j + \sum_{k \in C_C} y_k/a_k \leq \frac{1}{2}(|C_B| - |C_C| - 1) + \sum_{k \in C_C} u_k/a_k. \tag{13}$$

We can rewrite (13) as

$$\sum_{j \in C_B} (1 - 2x_j) + \sum_{k \in C_C} \left(\frac{2(u_k - y_k)}{a_k} - 1 \right) \geq 1. \tag{14}$$

Then, given $(\bar{x}, \bar{y})$, finding a most violated mixed odd cycle inequality is equivalent to finding a minimum weight odd cycle on a graph with edge weights

$$w(i,k) = \begin{cases} 1 - \bar{x}_i - \bar{x}_k, & \text{if } i, k \in N, \\ \frac{u_k - \bar{y}_k}{a_k} - \bar{x}_i, & \text{if } i \in N, k \in M. \end{cases}$$

Observe that for a point $(\bar{x}, \bar{y}) \in$ LMVP, $w(i,k) \geq 0$ for all $(i,k) \in E \cup F$. Since there is a polynomial time algorithm to find a minimum weight odd cycle on a graph with nonnegative edge weights [7], the separation problem is solvable in polynomial time. □

3.3 Strengthening star inequalities

In this section, we present a procedure for strengthening star inequalities when the binary variables appearing in the inequality are not independent. A strengthened star inequality has the form $\sum_{j \in T} \tilde{a}_{jk} x_j + y_k \leq u_k$ with $\tilde{a}_{jk} \geq \bar{a}_{jk}$ for $j \in T$. The procedure begins with a star inequality (7), and then the coefficients are increased iteratively in increasing order of $a_{ik}, i \in T$.

Proposition 3.12 *Let $\sum_{j \in T} \tilde{a}_{jk} x_j + y_k \leq u_k$ be a strengthened star inequality such that for some $i \in T$, $\tilde{a}_{jk} = \bar{a}_{jk}$ for $j \in T$ with $a_{jk} > a_{ik}$ and $\tilde{a}_{jk} \geq \bar{a}_{jk}$ for $j \in T$ with $a_{jk} \leq a_{ik}$. Then the coefficient of variable x_i can be increased by*

$$(a_{ik} - \sum_{j \in S} \tilde{a}_{jk})^+$$

where $S = \{j \in T \setminus N(i) : a_{jk} \leq a_{ik}\}$ and $a^+ = \max(a, 0)$.

Proof. Let δ_i denote the increase in the coefficient of x_i. For the inequality to remain valid for MVP, we need

$$\delta_i \leq u_k - \max_{(x,y) \in MVP, x_i = 1} \left\{ \sum_{j \in T} \tilde{a}_{jk} x_j + y_k \right\}. \tag{15}$$

Let $U \subseteq T$ be the binary variables that have value one in an optimal solution to the right hand side of (15). Furthermore, let $\bar{a} = \max_{j \in U} a_{jk}$. Then

$$\max_{(x,y) \in MVP, x_i = 1} \left\{ \sum_{j \in T} \tilde{a}_{jk} x_j + y_k \right\} \quad \leq \quad \sum_{j \in S} \tilde{a}_{jk} + \sum_{j \in U \setminus S} \tilde{a}_{jk} + u_k - \bar{a}.$$

Since $\sum_{j \in U \setminus S} \tilde{a}_{jk} \leq \bar{a} - a_{ik}$, the result follows. □

Example. Consider the mixed conflict graph given in Figure 7. One of the star inequalities is

$$x_1 + x_2 + 3x_3 + 2x_4 + y \leq 10.$$

Increasing the coefficients of the star inequality in the order 1, 2, 3, 4 we obtain

$$x_1 + 2x_2 + 3x_3 + 4x_4 + y \leq 10.$$

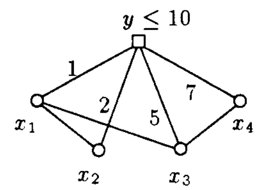

Figure 7: Strengthening star inequalities.

3.4 Sequential lifting

When the inequalities described previously are not facet-defining, we can make them stronger through lifting. We start with examining the lifting of a continuous variable. Let

$$\sum_{i \in S} \alpha_i x_i \leq r$$

be a valid inequality for MVP(N) and consider lifting a continuous variable y_k. Let α_k be the coefficient of y_k in the lifted inequality. In order for the inequality to be valid, we need

$$\alpha_k \quad \leq \quad \min \left\{ \frac{r - \sum_{i \in S} \alpha_i x_i}{y_k} : (x, y) \in MVP, y_k > 0 \right\}.$$

Proposition 3.13 *Let $\sum_{i \in S} \alpha_i x_i \leq r$ be a valid inequality for MVP(N). If S is a subset of $N(k)$ such that $a_{ik} = u_k$ for all $i \in S$ then*

$$\sum_{i \in S} \alpha_i x_i + \frac{r}{u_k} y_k \leq r$$

is a valid inequality for MVP.

Next, given a valid inequality of the form, $\sum_{i\in S}\alpha_i x_i + y_k \leq u_k$, $S \subseteq N(k)$, we consider lifting the binary variables in $N(k)\setminus S$. For $S \subseteq N(k)$ and $j \in N(k)\setminus S$, let $\mathcal{P}$ be the collection of packings that contain j in the graph induced by the vertex set $S \cup \{j\}$. Let

$$\sum_{i\in S}\alpha_i x_i + y_k \leq u_k$$

be a valid inequality for MVP and consider lifting binary variable $x_j \in N(k)\setminus S$. Let α_j be the coefficient of x_j in the lifted inequality. In order for the inequality to be valid, we need

$$\alpha_j \leq u_k - \max_{(x,y)\in MVP, x_j=1}\left\{\sum_{i\in S}\alpha_i x_i + y_k\right\},$$

or equivalently

$$\begin{aligned}\alpha_j &\leq u_k - \max_{P\in\mathcal{P}}\left\{\sum_{i\in P, i\neq j}\alpha_i + \min_{i\in P}\{u_k - a_{ik}\}\right\}\\ &= \min_{P\in\mathcal{P}}\left\{\max_{i\in P} a_{ik} - \sum_{i\in P, i\neq j}\alpha_i\right\}.\end{aligned}$$

Proposition 3.14

1. *If $S \subseteq N(j)$ then the maximum lifting coefficient of x_j, $\min_{P\in\mathcal{P}}\left\{\max_{i\in P} a_{ik} - \sum_{i\in P, i\neq j}\alpha_i\right\}$ is equal to a_{jk}.*
2. *For $j \in N(k)\setminus S$, if $a_{ik} \leq a_{jk}$ for all $i \in S$ then $a_{jk} - \max_{P\in\mathcal{P}}\sum_{i\in P, i\neq j}\alpha_i$ gives the maximum lifting coefficient.*

Now we give a class of *mixed odd wheel inequalities* that can be obtained by lifting a mixed odd cycle inequality. The proof of the next proposition is a simple application of the previous results on lifting.

Proposition 3.15 *Let $C = (C_B, C_C)$ be a mixed odd cycle. Then the mixed odd-wheel inequality*

$$\sum_{j\in C_B}(1 + \sum_{k\in M_j}\frac{a_k^2 - a_k^1}{a_k^1})x_j + \sum_{k\in C_C} y_k/a_k^1 + \alpha_w z_w \leq \lfloor\frac{|C_B| - |C_C|}{2}\rfloor + \sum_{k\in C_C} u_k/a_k^1 \tag{16}$$

is valid for conv(MVP), where a_k^1, a_k^2 are the weights of the incident edges to $k \in C_C$ in C, with $a_k^1 \leq a_k^2$ and $M_j = \{k \in M(j)\cap C_C : a_k^2 = a_{jk}\}$,

$$\alpha_w = \begin{cases}\lfloor\frac{|C_B|-|C_C|}{2}\rfloor + \sum_{k\in C_C}\frac{a_{wk}}{a_k^1} & \text{if } w \in N,\ C \subseteq N(w)\cup M(w),\\ \lfloor\frac{|C_B|-|C_C|}{2}\rfloor & \text{if } w \in M,\ C_B \subseteq N(w),\ a_{ji} = u_w \text{for all } j \in C_B.\end{cases}$$

The lifting coefficient in the mixed wheel inequality could be computed exactly due to the special structure of an odd cycle. In general, computing a lifting coefficient is hard. Therefore, we consider approximate lifting coefficients.

Proposition 3.16 *Let $\mathcal{P}$ be defined as before. Then $\left(a_{jk} - \max_{P\in\mathcal{P}} \sum_{i\in P, i\neq j} \alpha_i\right)^+$ is an approximation for the exact lifting coefficient.*

Proof. Decomposing the minimization problem of the lifting function we have

$$\begin{aligned}\min_{P\in\mathcal{P}} \left\{ \max_{i\in P} a_{ik} - \sum_{i\in P, i\neq j} \alpha_i \right\} &\geq \left(\min_{P\in\mathcal{P}} \max_{i\in P} a_{ik} - \max_{P\in\mathcal{P}} \sum_{i\in P, i\neq j} \alpha_i \right)^+ \\ &= \left(a_{jk} - \max_{P\in\mathcal{P}} \sum_{i\in P, i\neq j} \alpha_i \right)^+ .\end{aligned}$$

Equality follows since $j \in P$ for all $P \in \mathcal{P}$ by definition of $\mathcal{P}$. □

Proposition (3.16) suggests an easy way of generating valid inequalities by sequentially lifting $y_k \leq u_k$ with x_i, $i \in N(k)$. Let $i_1, i_2, \ldots, i_l$ be an arbitrary ordering of $N(k)$. Then

$$\sum_{j=1}^{l} \alpha_{i_j} x_{i_j} + y_k \leq u_k$$

is a valid inequality for *conv*(MVP) where the coefficients α_{i_j} are calculated as follows:

$$\alpha_{i_j} = \left(a_{i_j k} - \sum_{h\in S} \alpha_h \right)^+ \tag{17}$$

where $S = \{i_1, i_2, \ldots, i_{j-1}\} \setminus N(i_j)$. We call inequalities generated this way *lifted bound inequalities.* Note that both star and mixed clique inequalities are special cases of the lifted bound inequalities. We obtain a star inequality when we assume $N(i) = \emptyset$ for all $i \in N(k)$ and we obtain a mixed clique inequality when $S = \emptyset$. Lifted bound inequalities may be stronger than star and mixed clique inequalities in case the star and mixed clique inequalities are not facet-defining. Also, observe that a strengthened star inequality is a lifted bound inequality.

By exploiting the structure of $G(S)$, one can clearly, derive stronger lifted bound inequalities. For example, if $G(S)$ is a clique, then $\alpha_{i_j} = a_{i_j k} - \max_{h\in S} \alpha_h$. A simple modification to (17) allows us not only to derive stronger lifted bound inequalities, but also to generate lifted mixed clique inequalities. Let $G(K)$ be a clique of $G(S)$, then

$$\bar{a}_{i_j k} = \left(a_{i_j k} - \sum_{h\in S\setminus K} \alpha_h - \max_{h\in K} \alpha_h \right)^+ . \tag{18}$$

Note that if K is a singleton, the inequality is a regular lifted bound inequality.

4 Computational experiments

In order to test the effectiveness of the valid inequalities derived in the previous section, we implemented a branch-and-cut algorithm using MINTO [9] (version 3.0). MINTO is a customizable software system that solves mixed-integer linear programs by a branch-and-bound algorithm with linear programming relaxations. We performed computational experiments on two data sets. The first set consists of randomly generated mixed vertex packing problems. The second set consists of mixed integer problems from MIPLIB [3] for which violated star inequalities are generated. All experiments were done on an IBM RS/6000 Model 590 workstation with one hour CPU time limit.

Our results on the mixed vertex packing problems are summarized in Table 1. Clique inequalities on binary variables are certainly valid for MVP and MINTO generates them automatically. To see the effect of the new inequalities, we compared the performance of a branch-and-cut algorithm with clique, star and lifted bound inequalities against one with only clique inequalities on randomly generated graphs with varying edge density and fraction of continuous vertices. We used the algorithm given in Section 3.2 to separate the star inequalities. Once a violated star inequality is found, it is strengthened as explained in Section 3.3. Given a fractional solution $(\bar{x}, \bar{y})$, for each continuous variable y_k, the lifted bound inequalities are generated by lifting its adjacent binary variables x_i in nonincreasing order of $a_{ik}\bar{x}_i$. If the resulting inequality is violated by the fractional solution, it is added to the formulation. In Table 1, for each case we give the average duality gap ($\text{LPgap} = 100 \times \frac{\text{zroot}-\text{zopt}}{\text{zopt}}$) at the root node after all the valid inequalities are added, the number of inequalities generated, the number of nodes explored, and the total CPU time elapsed in seconds of five instances with 100 and 150 vertices. Observe that as the fraction of continuous variables increases, MVPPs become easier to solve. Problems with 20% continuous vertices could not be solved to optimality for densities 0.2 and 0.4 and 150 vertices by either algorithm. However, the duality gap is reduced considerably with the addition of the new inequalities. For these problems, since optimal solutions are not known, we use the best incumbent solution to report the duality gap at the root node. We remark that in both cases the best incumbent solutions are found when star and lifted bound inequalities are added. From these results, we conclude that the star inequalities and the lifted bound inequalities are effective in strengthening the LP relaxations and in reducing the number of nodes explored and the overall solution times.

vertices	density	cont	LPgap	clqs	nodes	time	LPgap	clqs	stars	lft bnds	nodes	time
100	0.1	0.2	9.66	66	135	32	4.90	59	113	1	28	27
	0.1	0.4	9.59	38	327	38	2.29	28	148	3	8	16
	0.2	0.2	26.13	693	1380	277	16.57	299	530	14	152	170
	0.2	0.4	13.39	69	77	26	0.47	35	160	24	2	8
	0.4	0.2	46.86	1012	978	476	21.96	462	482	109	105	307
	0.4	0.4	6.23	64	26	25	0.00	20	29	40	1	5
150	0.1	0.2	22.28	1009	5970	2811	16.47	502	480	16	1950	2365
	0.1	0.4	11.25	166	1308	492	2.41	56	378	9	13	65
	0.2	0.2	50.66	3481	5120	3600	32.42	1652	2964	51	1384	3600
	0.2	0.4	7.52	78	43	49	0.00	30	76	44	1	8
	0.4	0.2	61.54	3161	1610	3600	30.47	1154	1184	308	255	3600
	0.4	0.4	4.76	92	55	91	0.00	20	46	64	1	21

Table 1: Performance measures for MVPPs.

Results on the second data set are presented in Table 2. In this table, we report the number of star cuts generated, the duality gap at the root node, the percentage gap between the best upper bound and the best lower bound at termination (endgap $= 100 \times \frac{\text{zub}-\text{zlb}}{\text{zlb}}$), the number of nodes explored in the search tree and the total CPU time elapsed in seconds. The addition of the star cuts reduces the duality gap for the majority of the problems and decreases the number of nodes explored for almost all the problems. Note that even for problems where there is no or modest reduction in the duality gap, the performance of the algorithm improves after adding cuts. For a few problems, even though the number of nodes explored decreased, the overall solution time increased. This is because the addition of new constraints yields longer LP solution times. Five problems could not be solved within an hour of CPU time; however for the hardest two of those, **mod011** and **set1ch**, the gap between upper and lower bounds at termination is reduced significantly. For **set1ch**, we use the already known optimal value to report the endgap, because no feasible solution was found by either of the algorithms within one hour of CPU time. We note that no violated lifted bound inequalities were found for the MIPLIB problems.

	without star cuts				with star cuts				
problem	LPgap	endgap	nodes	time	cuts	LPgap	endgap	nodes	time
bell3a	1.40	0.00	50347	531	4	1.39	0.00	38683	361
blend2	8.99	0.00	5535	105	112	8.99	0.00	4790	110
dcmulti	1.92	0.00	6569	127	36	1.46	0.00	5045	125
egout	9.94	0.00	205	1	124	5.64	0.00	21	1
fixnet4	13.79	0.00	1056	44	7	13.68	0.00	642	40
gen	0.04	0.00	529	24	26	0.04	0.00	182	13
gesa2	1.11	0.99	47137	3600	3	1.11	1.24	44855	3600
gesa2_o	1.12	1.25	41752	3600	9	1.12	1.18	35004	3600
gesa3	0.52	0.00	3409	308	6	0.52	0.00	3860	413
gesa3_o	0.52	0.00	7962	705	7	0.52	0.00	2049	267
khb05250	10.31	0.00	3605	109	98	0.81	0.00	13	3
mod011	13.86	12.52	1126	3600	2200	7.35	6.06	766	3600
qnet1	10.95	0.00	205	48	11	10.95	0.00	173	48
qnet1_o	19.48	0.00	449	61	9	15.69	0.00	119	31
rentacar	5.11	0.00	131	338	67	5.11	0.00	107	399
rgn	40.63	0.00	2655	16	20	38.81	0.00	2661	18
rout	8.88	8.65	73702	3600	31	8.88	8.83	38419	3600
set1ch	35.61	33.19	25957	3600	148	25.45	25.45	18600	3600
vpm2	18.11	0.00	195577	2815	16	16.96	0.00	1844387	3110

Table 2: Performance measures for the MIPLIB problems.

5 Extensions

The reduction given in Section 2 does not preserve perfection, in general. A graph is said to be *chordal* if every cycle of length greater than or equal to four has a *chord*, that is, an edge incident to two nonconsecutive vertices of the cycle. A graph is *perfect* if the size of the largest clique (*clique number*) is equal to the smallest number of vertex packings needed to cover the vertices (*chromatic number*) for all induced subgraphs of the graph. Both chordal graphs and comparability graphs are subclasses of perfect graphs [6]. Perfect graphs are one of the largest classes of graphs, on which the vertex packing problem is known to be solvable in polynomial time [7]. However, neither chordal nor comparability graphs are mapped to perfect graphs by the reduction. In Figure 8, G is a chordal as well as a comparability graph, but G' is not perfect since it has an odd hole (drawn with dashed edges). Nevertheless, we conjecture that the mixed vertex packing problem is solvable in polynomial time on these graphs and possibly on all classes of perfect graphs. We are currently working on a primal-dual algorithm for the case of independent binary vertices.

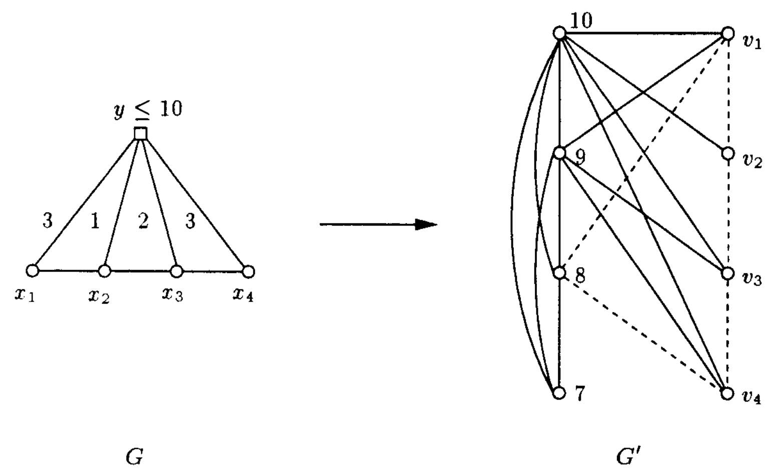

Figure 8: Chordal and comparability graphs.

With the insights we gained from studying MVPP, we are also investigating how to use the mixed vertex packing relaxation together with a single mixed integer knapsack inequality to obtain stronger relaxations for general 0-1 MIP problems. It is well-known that any mixed integer inequality can be relaxed to one with only continuous variables. Since a variable upper bound constraint is a special case of a mixed edge inequality, we can derive generalizations of the flow cover inequalities [15] in this manner. To be more precise, we are interested in deriving strong valid inequalities for

$$\sum_{i\in M^+} y_i - \sum_{i\in M^-} y_i \leq b$$

$$x_i + x_j \leq 1, \quad (i,j) \in E$$

$$a_{ik}x_i + y_k \leq u_k, \quad (i,k) \in F.$$

References

[1] A. Atamtürk. *Conflict graphs and flow models for linear mixed-integer optimization problems.* PhD thesis, ISyE, Georgia Institute of Technology, Atlanta, USA, 1998.

[2] A. Atamtürk, G. L. Nemhauser, and M. W. P. Savelsbergh. Conflict graphs in integer programming. Technical Report TLI-98-03, ISyE, Georgia Institute of Technology, 1998.

[3] R. E. Bixby, S. Ceria, C. M. McZeal, and M. W. P. Savelsbergh. An updated mixed integer programming library: MIPLIB 3.0. Technical Report TR98-03, Department of Computational and Applied Mathematics, Rice University, 1996. Available from URL `http://www.caam.rice.edu/~bixby/miplib/miplib.html`.

[4] R. Borndörfer and R. Weismantel. Set packing relaxations of some integer programs. Technical Report SC 97-30, Konrad-Zuse-Zentrum für Informationstechnik Berlin, 1997.

[5] E. Cheng and W. H. Cunningham. Wheel inequalities for stable set polytopes. *Mathematical Programming*, 77, 389–421, 1997.

[6] M. Golumbic. *Algorithmic Graph Theory and Perfect Graphs.* Academic Press, New York, 1980.

[7] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization.* Springer-Verlag, 1993.

[8] E. L. Johnson. Modeling and strong linear programs for mixed integer programming. In S. W. Wallace, editor, *Algorithms and Model Formulations in Mathematical Programming*, pages 1–43. Springer-Verlag, 1989. NATO ASI Series, Vol. F51.

[9] G. L. Nemhauser, M. W. P. Savelsbergh, and G. S. Sigismondi. MINTO, a Mixed INTeger Optimizer. *Operations Research Letters*, 15, 47–58, 1994.

[10] G. L. Nemhauser and G. Sigismondi. A strong cutting plane/branch-and-bound algorithm for node packing. *Journal of the Operational Research Society*, 43, 443–457, 1992.

[11] G. L. Nemhauser and L. E. Trotter Jr. Properties of vertex packing and independence system polyhedra. *Mathematical Programming*, 6, 48–61, 1974.

[12] M. Padberg. On the facial structure of set packing polyhedra. *Mathematical Programming*, 5, 199–215, 1973.

[13] M. W. P. Savelsbergh. Preprocessing and probing techniques for mixed integer programming problems. *ORSA Journal on Computing*, 6, 445–454, 1994.

[14] L. E. Trotter Jr. A class of facet producing graphs for vertex packing polyhedra. *Discrete Mathematics*, 12, 373–388, 1975.

[15] T. J. Van Roy and L. A. Wolsey. Valid inequalities for mixed 0-1 programs. *Discrete Applied Mathematics*, 14, 199–213, 1986.