operations research letters

The partial constraint satisfaction problem: Facets and lifting theorems

Arie M.C.A. Koster, Stan P.M. van Hoesel *, Antoon W.J. Kolen

Department of Quantitative Economics, Maastricht University, P.O. Box 616, 6200 MD Maastricht, The Netherlands

Received 1 March 1997; received in revised form 1 July 1998

Abstract

We study the polyhedral structure of the partial constraint satisfaction problem (PCSP). Among the problems that can be formulated as such are the maximum satisfiability problem and a fairly general model of frequency assignment problems. We present lifting theorems and classes of facet defining inequalities, and we provide preliminary experiments. © 1998 Elsevier Science B.V. All rights reserved.

Keywords: Integer programming; Constraint satisfaction; Facets; Lifting

1. The partial constraint satisfaction problem

A partial constraint satisfaction problem (PCSP) is defined by a so-called constraint graph $G=(V,E)$. Each vertex $v \in V$ in this graph represents a decision variable, that can obtain a value from a given domain D_v. Each value has a penalty attached to it. Moreover, an edge $\{v,w\} \in E$ in the graph indicates that some combinations of domain elements of v and w are also penalized. The objective of the PCSP is to select a domain element for each vertex such that the total penalty incurred is minimized. More formally, a PCSP is defined by the quadruple $(G=(V,E), D_V, P_E, Q_V)$. D_V is a set of domains D_v, $v \in V$ where each domain is a finite set. P_E is a set of edge-penalty functions $P_{\{v,w\}}: \{\{d_v,d_w\} \mid d_v \in D_v, d_w \in D_w\} \to \mathbb{R}$, $\{v,w\} \in E$, and Q_V is a set of vertex-penalty functions $Q_v: D_v \to \mathbb{R}$, $v \in V$. The objective is to minimize the total sum of the penalties $\sum_{\{v,w\}\in E} P_{\{v,w\}}(\{d_v,d_w\}) + \sum_{v\in V} Q_v(d_v)$.

Many frequency assignment problems (FAP) described in the literature belong to the class of partial constraint satisfaction problems. For example, in the FAP in which we have to assign a frequency to each transceiver in a mobile telephone network, a vertex corresponds to a transceiver. The domain of a vertex is the set of frequencies that can be assigned to that transceiver. An edge indicates that communication from one transceiver may interfere with communication from the other transceiver. In most applications interference occurs whenever the distance between the frequencies assigned to the transceivers is less than a given threshold depending on the two transceivers. The penalty of an edge reflects the priority with which interference should be avoided, whereas the penalty on a vertex can be seen as a level of preference for the frequencies.

* Corresponding author. Fax: +31-43 325 8535; e-mail: s.vanhoesel@ke.unimaas.nl.

PII: S0167-6377(98)00043-1

For another type of frequency assignment problems, involving receiver–transmitter pairs of radio links, that can be formulated as a partial constraint satisfaction problem, we refer to Kolen [3]. Forthe special case in which no interference is allowed (is there a solution with penalty zero) the polyhedral structure of the problem is studied in Aardal et al. [1].

The maximum satisfiability problem (MAX SAT) can be reformulated elegantly as a partial constraint satisfaction problem. In a MAX SAT problem m clauses $c_1,\dots,c_m$ involving the boolean variables $x_1,\dots,x_n$ are given. Each clause contains a number of literals, where a literal is either a variable or the negation of a variable. The problem is to assign a value true or false to each variable so as to maximize the number of clauses that are satisfied. A clause is satisfied if at least one literal in it has the value true.

To model MAX SAT as a PCSP, we introduce a vertex v_{c_i} for every clause c_i, $i=1,\dots,m$, and a vertex v_{x_j} for every variable x_j, $j=1,\dots,n$. The domain of v_{c_i} contains a element for each literal in the clause c_i; let us denote this element by the literal itself. The domain of v_{x_j} is given by $\{\mathit{true},\mathit{false}\}$. There is an edge between a vertex v_{c_i} representing clause c_i, and a vertex v_{x_j} representing variable x_j if and only if $x_j\in c_i$ or $\bar{x}_j\in c_i$ ($\bar{x}_j$ is the negation of x_j). If $x_j\in c_i$, then the penalty of the combination of domain values (x_j,false) is equal to 1. If $\bar{x}_j\in c_i$, then the penalty of the combination of domain values $(\bar{x}_j,\mathit{true})$ is equal to 1. All other penalties are zero. The optimal value of this partial constraint satisfaction problem is k if and only if the optimal value of the corresponding MAX SAT is $m-k$. Furthermore, an optimal solution of the MAX SAT is given by the domain values selected for the vertices corresponding to the variables in the optimal solution of the partial constraint satisfaction problem. This shows that the two problems are equivalent. Since MAX 2 SAT (each clause contains at most 2 literals) is NP-hard [2] a partial constraint satisfaction problem with $|D_v|=2$ for all $v\in V$ is already NP-hard.

For the MAX 2 SAT problem a more compact formulation is possible. We have a vertex v_{x_j} corresponding to every variable x_j, and the domain is given by $\{\mathit{true},\mathit{false}\}$. There is an edge $\{v_{x_i},v_{x_j}\}$ if and only if there exists a clause containing a literal corresponding to x_i and a literal corresponding to x_j. The penalty corresponding to a combination of values for the variables x_i and x_j is equal to the number of clauses containing literals corresponding to both variables for which the given combination does not satisfy the clause.

The satisfiability problem (SAT), in which the question is whether there is an assignment of the variables for which all clauses are satisfied, can also be formulated as a partial constraint satisfaction problem as follows. There is one vertex for every clause and an edge if the two corresponding clauses contain a conflicting literal corresponding to the same variable. A combination $\{x_i,\bar{x}_i\}$ with $x_i\in C_j$ and $\bar{x}_i\in C_k$ has penalty one. All combinations corresponding to non-conflicting literals have penalty zero. A problem instance is satisfiable if and only if the corresponding partial constraint satisfaction problem instance has optimal value zero.

The PCSP can be viewed as a linearization of the boolean quadric polytope (see [6]) and is therefore related to the transitive packing polytope (see [4]).

In Section 2 of this paper we formulate the partial constraint satisfaction problem as $\{0,1\}$ linear programming problem, we state the dimension of the problem, and describe trivial facet defining valid inequalities. We prove theorems for lifting facets of a subproblem to facets for the original problem in Section 3. In Section 4 we define some classes of facets for the PCSP. Some preliminary computational results are addressed in Section 5, whereas the last section contains the concluding remarks.

2. Formulation, dimension and trivial facets

To formulate the partial constraint satisfaction problem as a $\{0,1\}$-programming problem we introduce the following $\{0,1\}$-variables for all $v\in V$, $d_v\in D_v$

$$y(v,d_v)=\begin{cases}1 & \text{if } d_v\in D_v \text{ is selected,}\\ 0 & \text{otherwise}\end{cases}$$

and for all $\{v,w\}\in E$, $d_v\in D_v$, $d_w\in D_w$

$$z(v,d_v,w,d_w)=\begin{cases}1 & \text{if } (d_v,d_w)\in D_v\times D_w\\ & \text{is selected,}\\ 0 & \text{otherwise.}\end{cases}$$

In the sequel, let $q(v,d_v)$ and $p(v,d_v,w,d_w)$ denote $Q_v(d_v)$ and $P_{\{v,w\}}(\{d_v,d_w\})$, respectively.

A $\{0,1\}$-programming formulation of the partial constraint satisfaction problem is given by

$$\min \quad \sum_{\{v,w\}\in E}\sum_{d_v\in D_v}\sum_{d_w\in D_w} p(v,d_v,w,d_w)z(v,d_v,w,d_w) + \sum_{v\in V}\sum_{d_v\in D_v} q(v,d_v)y(v,d_v) \tag{1}$$

$$\text{s.t.} \quad \sum_{d_v\in D_v} y(v,d_v)=1 \quad \forall v\in V, \tag{2}$$

$$\sum_{d_w\in D_w} z(v,d_v,w,d_w) = y(v,d_v) \quad \forall\{v,w\}\in E,\ d_v\in D_v, \tag{3}$$

$$z(v,d_v,w,d_w)\in\{0,1\} \quad \forall\{v,w\}\in E,\ d_v\in D_v,\ d_w\in D_w, \tag{4}$$

$$y(v,d_v)\in\{0,1\} \quad \forall v\in V,\ d_v\in D_v. \tag{5}$$

Constraints (2) model the fact that exactly one value in the domain of a vertex should be selected. Constraints (3) enforce that the combination of values selected for an edge should be consistent with the values selected for the vertices of that edge.

We define the partial constraint satisfaction polytope $X(\mathrm{PCSP})$ to be the convex hull of all $\{0,1\}$-vectors (y,z) satisfying Eqs. (2) and (3). Although the y-variables can be eliminated from the formulation, we believe that it is more convenient to keep them in the formulation. Note that once the y-variables are $\{0,1\}$ the z-variables are forced to be integral.

The dimension of the partial constraint satisfaction polytope is given by Theorem 2.1.

Theorem 2.1. *The dimension of* $X(\mathrm{PCSP})$, *defined by* $(G=(V,E),D_V)$ *is*

$$\sum_{v\in V}(|D_v|-1)+\sum_{\{v,w\}\in E}(|D_v|-1)(|D_w|-1). \tag{6}$$

Proof. We will first prove that $X(\mathrm{PCSP})$ satisfies $|V|+\sum_{\{v,w\}\in E}(|D_v|+|D_w|-1)$ (number of variables minus dimension) linearly independent equalities, which implies that Eq. (6) is an upper bound for the dimension. These linear independent equalities are obtained by taking the $|V|$ constraints (2), and for every edge $\{v,w\}$ all but one $(=\sum_{\{v,w\}\in E}(|D_v|+|D_w|-1))$ of the constraints (3). Note that constraints (3) for a given edge $\{v,w\}$ can be viewed as the constraints of a transportation problem with suppliers indicated by (v,d_v) with supply $y(v,d_v)$ and clients indicated by (w,d_w) with demand $y(w,d_w)$. Thus, deleting one of these constraints results in a set of linear independent equalities.

Next, we will prove that Eq. (6) is a lower bound for the dimension by supplying $1+\sum_{v\in V}(|D_v|-1)+\sum_{\{v,w\}\in E}(|D_v|-1)(|D_w|-1)$ affinely independent feasible solutions. Note that once the y-variables are given, the z-variables are uniquely determined by constraints (3). To define these solutions we arbitrarily select a value $d_v^*\in D_v$. A first solution is given by $y(v,d_v^*)=1$ for all $v\in V$.

Next, we construct $\sum_{v\in V}(|D_v|-1)$ solutions which differ from the first solution in only one domain element: for each $v\in V$, $d_v\in D_v\setminus\{d_v^*\}$, we define the solution $y(v,d_v)=1$, $y(w,d_w^*)=1$ for all $w\neq v$. Lastly, we construct $\sum_{\{v,w\}\in E}(|D_v|-1)(|D_w|-1)$ solutions which differ from the first solution in two domain elements of adjacent vertices: for each $\{v,w\}\in E$, $d_v\in D_v\setminus\{d_v^*\}$, and $d_w\in D_w\setminus\{d_w^*\}$, we define the solution $y(v,d_v)=y(w,d_w)=1$ and $y(u,d_u^*)=1$ for all $u\in V\setminus\{v,w\}$. Note that all these solutions are affinely independent. □

The following theorem shows that many of the trivial inequalities are facet defining.

Theorem 2.2. *For every* $\{v,w\}\in E$, $|D_v|\geqslant 2$, $|D_w|\geqslant 2$, $d_v\in D_v$, $d_w\in D_w$ *the inequality*

$$z(v,d_v,w,d_w)\geqslant 0 \tag{7}$$

defines a facet for $X(\mathrm{PCSP})$.

Proof. Among the affinely independent solutions in the proof of the previous theorem, all solutions but one satisfy Eq. (7) with equality. □

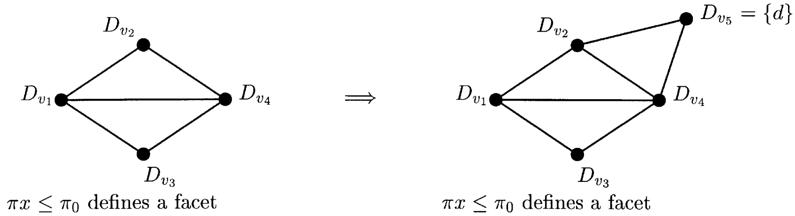

Fig. 1. Extension of the graph.

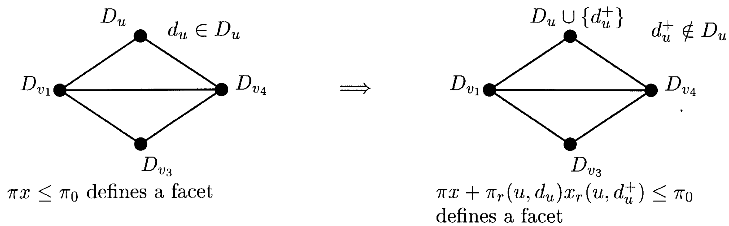

Fig. 2. Extension of the domain.

3. Lifting theorems

In this section we will discuss two types of lifting. Combining them enables us to lift a facet defining inequality of a particular PCSP to facet defining inequalities for an extended PCSP. First, we show that a facet defining inequality remains facet defining if the constraint graph is extended with vertices having one domain element (see Fig. 1). Second, we show how a facet defining inequality can be extended if the domain of a vertex is extended with copies of other domain elements (see Fig. 2).

If X(PCSP) is defined by $(G=(V,E),D_V)$, let X_u(PCSP) denote the PCSP-polytope defined by the extended graph on $G_u=(V\cup\{u\},E\cup E_u)$ where E_u is the set of edges incident to u, with the same domains for $v\in V$ and $|D_u|=1$. Moreover, let $x=(y,z)$ denote the solution vector.

Theorem 3.1. *Let* X(PCSP) *be defined by* $(G=(V,E),D_V)$. *If* $\pi x\leqslant\pi_0$ *is a facet defining inequality for* X(PCSP), *then* $\pi x\leqslant\pi_0$ *is a facet defining inequality for* X_u(PCSP).

Proof. The polytope X(PCSP) is a projection of X_u(PCSP) and both have the same dimension (see Theorem 2.1). $\square$

Next, we show how a facet defining inequality of a PCSP defined by the constraint graph $G=(V,E)$ and a set of domains D_v, $v\in V$, can be lifted into a facet defining inequality for the PCSP. This is done by using the same constraint graph and set of domains D_v^+, $v\in V$, where $D_v^+=D_v$, for all $v\in V$, $v\neq u$, and $D_u^+=D_u\cup\{d_u^+\}$ (see Fig. 2). Theorem 3.2 states that if we give each variable related to d_u^+ the same coefficient as the corresponding variable of an arbitrarily selected domain element $d_u\in D_u$, then the new inequality is facet defining for the extended problem whenever the original inequality is facet defining for the original problem. In order to prove Theorem 3.2 we need the following two lemmas.

Lemma 3.1. *Let* $u\in V$, $d_u\in D_u$, *and let* δ_u *define the set of neighbors of* u, $\delta_u=\{v\mid\{u,v\}\in E\}$. *If*

$$\beta(u,d_u)y(u,d_u)+\sum_{v\in\delta_u}\sum_{d_v\in D_v}\gamma(v,d_v)z(u,d_u,v,d_v)\geqslant 0 \tag{8}$$

is a facet defining inequality for X(PCSP), *then the inequality describes a trivial facet.*

Proof. Let $d_v^*=\operatorname{argmin}_{d_v\in D_v}\gamma(v,d_v)$ for all $v\in\delta_u$. Adding $\gamma(v,d_v^*)$ times the model equality

$y(u,d_u) - \sum_{d_v \in D_v} z(u,d_u,v,d_v) = 0$ to Eq. (8) for all $v \in \delta_u$ results in the inequality

$$\left[\beta(u,d_u) + \sum_{v\in\delta_u} \gamma(v,d_v^*)\right] y(u,d_u)$$
$$+ \sum_{v\in\delta_u} \sum_{d_v\in D_v} [\gamma(v,d_v) - \gamma(v,d_v^*)] z(u,d_u,v,d_v) \geqslant 0$$

or using $y(u,d_u) = \sum_{d_{v'}\in D_{v'}} z(u,d_u,v',d_{v'})$ for a specific $v' \in \delta_u$

$$\left[\beta(u,d_u) + \sum_{v\in\delta_u} \gamma(v,d_v^*)\right] \sum_{d_{v'}\in D_{v'}} z(u,d_u,v',d_{v'})$$
$$+ \sum_{v\in\delta_u} \sum_{d_v\in D_v} [\gamma(v,d_v) - \gamma(v,d_v^*)] z(u,d_u,v,d_v) \geqslant 0 \tag{9}$$

The validity of Eq. (8) implies that $\beta(u,d_u) + \sum_{v\in\delta_u} \gamma(v,d_v^*) \geqslant 0$. Thus all coefficients of Eq. (9) are non-negative. Furthermore, at least one coefficient is positive, otherwise Eq. (9) is a linear combination of the model equalities. Hence, the face defined by Eq. (8) is a subset of a trivial facet and thus, it can only be a trivial facet. □

In the sequel, for a given (u,d_u), we use $x_r(u,d_u)$ $(\pi_r(u,d_u))$ as the restriction of the vector $x(\pi)$ to the components related to (u,d_u), i.e., the variables $y(u,d_u)$ and $z(u,d_u,v,d_v)$ for all $v \in \delta_u$.

Lemma 3.2. *Let $\pi x \leqslant \pi_0$ define a non-trivial facet of X(PCSP). Then for each (u,d_u), there are exactly $1 + \sum_{v\in\delta_u}(|D_v|-1)$ solutions with $y(u,d_u)=1$, $\pi x = \pi_0$, and for which $x_r(u,d_u)$ are affinely independent.*

Proof. Let $x^1,\ldots,x^p$ be $p = \dim X$(PCSP) affinely independent solutions which satisfy $\pi x \leqslant \pi_0$ with equality. Moreover, let $x^1,\ldots,x^q$ be q solutions with $y(u,d_u)=1$ which are affinely independent with respect to the components $y(u,d_u)$ and $z(u,d_u,v,d_v)$ for all $v \in \delta_u$, $d_v \in D_v$ ($x_r^1(u,d_u),\ldots,x_r^q(u,d_u)$ are affinely independent). Then we have to prove that $q = 1 + \sum_{v\in\delta_u}(|D_v|-1)$. Since $x_r^1(u,d_u),\ldots,x_r^q(u,d_u)$ all satisfy $y(u,d_u)=1$ these vectors are also linearly independent. So, it is sufficient to prove that the matrix $[x_r^1(u,d_u),\ldots,x_r^q(u,d_u)]$ with $1 + \sum_{v\in\delta_u}|D_v|$ rows has rank $1 + \sum_{v\in\delta_u}(|D_v|-1)$. Or, equivalently, it is sufficient to prove that the dimension of the row nullspace is $|\delta_u|$ (number of rows minus the rank of the matrix).

First, we prove that the dimension of the row nullspace is at least $|\delta_u|$. Every solution satisfies the model equalities $y(u,d_u) - \sum_{d_v\in D_v} z(u,d_u,v,d_v) = 0$ for all $v \in \delta_u$. So, if $\alpha^v = (\beta^v,\gamma^v)$ corresponds to the coefficients in the left-hand side of the equality for $v \in \delta_u$, then $\alpha^v x_r^i(u,d_u) = 0$ for $i = 1,\ldots,q$. Moreover, α^v, for $v \in \delta_u$ are linearly independent, which implies that the dimension of the row nullspace is at least $|\delta_u|$.

Now, suppose the dimension of the row nullspace is at least $|\delta_u| + 1$. Then there exists another non-zero vector $\alpha = (\beta,\gamma)$ with $\alpha x_r^i(u,d_u) = 0$ for all $i = 1,\ldots,q$ which is linearly independent from the vectors α^v, $v \in \delta_u$. For $j = q+1,\ldots,p$, either $y^j(u,d_u) = z^j(u,d_u,v,d_v) = 0$ or $x_r^j(u,d_u)$ is affinely dependent of $x_r^1(u,d_u),\ldots,x_r^q(u,d_u)$. Hence, these solutions also satisfy $\alpha x_r(u,d_u) = 0$. As a consequence, the facet described by $\pi x = \pi_0$ is a subset of the face described by $\alpha x_r(u,d_u) = 0$, i.e. $F := \{x \in X(\text{PCSP}) \mid \pi x x = \pi_0\} \subseteq \{x \in X(\text{PCSP}) \mid \alpha x_r(u,d_u) = 0\} =: F_\alpha$. If equality does not hold, then (since $\pi x \leqslant \pi_0$ describes a facet) $F_\alpha \equiv X$(PCSP) and $\alpha x_r(u,d_u) = 0$ is an implicit equality. However, α is linearly independent from the implicit equalities involving (u,d_u). Hence $F_\alpha \equiv F$. From Nemhauser and Wolsey [5] (Theorem 3.6, page 91) it follows that either $\alpha x_r(u,d_u) \geqslant 0$ or $-\alpha x_r(u,d_u) \geqslant 0$ is a valid inequality for X(PCSP) defining the same facet as $\pi x \leqslant \pi_0$. By Lemma 3.1, however, $\alpha x_r(u,d_u) \geqslant 0$ (or $-\alpha x_r(u,d_u) \geqslant 0$) describes a trivial facet, a contradiction. Consequently, the dimension of the row nullspace is exactly $|\delta_u|$. □

Now, we can prove the main theorem of this paper.

Theorem 3.2. *Let X(PCSP) be defined by $(G=(V,E),D_V)$. Let $u \in V$, $d_u \in D_u$. Define X^+(PCSP) by $(G=(V,E),D_V^+)$ with $D_v^+ = D_v$, $v \in V\backslash\{u\}$, $D_u^+ = D_u \cup \{d_u^+\}$. If $\pi x \leqslant \pi_0$ is a non-trivial facet defining inequality for X(PCSP), then*

$$\pi x + \pi_r(u,d_u) x_r(u,d_u^+) \leqslant \pi_0 \tag{10}$$

is facet defining for X^+(PCSP).

Proof. First, note that $\dim X^{+}(\text{PCSP}) = \dim X(\text{PCSP}) + 1 + \sum_{v\in\delta_u}(|D_v| - 1)$. Let the solutions $x^1,\dots,x^p$, where $p = \dim X(\text{PCSP})$, be a set of affinely independent solutions which satisfy $\pi x \leqslant \pi_0$ with equality. It follows from Lemma 3.2 that there exist $1 + \sum_{v\in\delta_u}(|D_v| - 1)$ solutions which satisfy $y(u,d_u) = 1$ and for which the restrictions to (u,d_u) are affinely independent. Replace in these solutions d_u by d_u^{+}. Then these new solutions together with the old solutions are affinely independent. □

4. Non-trivial classes of facet defining inequalities

In this section we introduce two classes of facet defining inequalities for the PCSP. The facets are characterized by an induced subgraph $G_S = (S, E_S)$ of the constraint graph $G = (V,E)$. For every $v \in S$ the domain D_v is partitioned into A_v and B_v. Domain values in A_v can be seen as copies of one another (i.e., their related variables have the same coefficients in the inequality); likewise the domain values in B_v. Therefore, the facet-proofs for these classes can be restricted to G_S and domains of size 2 (for all $v \in S$), which suffices according to the theorems of Section 3.

For notational convenience, we introduce

$$y(v,D_v') = \sum_{d_v\in D_v'} y(v,d_v) \quad \text{and}$$

$$z(v,D_v',w,D_w') = \sum_{d_v\in D_v'}\sum_{d_w\in D_w'} z(v,d_v,w,d_w)$$

for $D_v' \subseteq D_v$ and $D_w' \subseteq D_w$.

4.1. The cycle-inequalities

First, we introduce the *cycle-inequalities*. Let the induced subgraph $G_S = (S,E_S)$ of $G = (V,E)$ be a chordless k-cycle (i.e. $S = \{v_i \mid i = 1,\dots,k\}$, $E_S = \{\{v_i, v_{i+1}\}, \mid i = 1,\dots,k-1\}\} \cup \{\{v_k, v_1\}\}$), then a k-cycle inequality, $k \geqslant 3$, is given by

$$\sum_{i=1}^{k-1}(z(v_i,A_{v_i},v_{i+1},A_{v_{i+1}}) + z(v_i,B_{v_i},v_{i+1},B_{v_{i+1}})) + z(v_0,A_{v_0},v_k,B_{v_k}) + z(v_0,B_{v_0},v_k,A_{v_k}) \leqslant k-1. \tag{11}$$

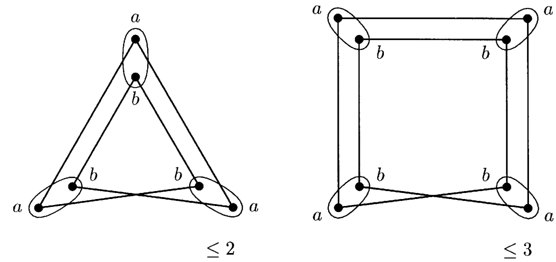

Fig. 3. Cycle inequalities.

Fig. 3 shows a 3-cycle inequality and a 4-cycle inequality. The *a-dot* represents the A-subset of the domain; the *b-dot* represents the B-subset of the domain. A line between two dots indicates that the coefficient corresponding to the indicated subsets is equal to one.

Theorem 4.1. *The k-cycle inequalities, $k \geqslant 3$, are valid and facet defining for X(PCSP).*

Proof. By the results of Section 3 it is sufficient to prove that the k-cycle inequalities are valid and facet defining for X(PCSP) defined by the k-cycle constraint graph and $A_{v_i} = \{a_{v_i}\}$, $B_{v_i} = \{b_{v_i}\}$, $i = 1,\dots,k$.

Consider an arbitrary solution x. Each edge of the cycle in the constraint graph contributes at most one to the left-hand side of Eq. (11). So, if at least one edge does not contribute to the left-hand side, Eq. (11) is satisfied by x. If all edges $\{v_i, v_{i+1}\}$ for $i = 1,\dots,k-1$ contribute 1 to the left-hand side, then either a_{v_i} is selected, for $i = 1,\dots,k$ or b_{v_i} is selected, for $i = 1,\dots,k$. But, then the edge $\{v_k, v_1\}$ does not contribute to the left-hand side. Hence, x satisfies Eq. (11).

A k-cycle inequality is satisfied with equality if exactly one edge of the cycle does not contribute 1 to the left-hand side. The k solutions ($j \in \{1,\dots,k\}$) in which a_{v_i} is selected for $1 \leqslant i \leqslant j$ and b_{v_i} for $j+1 \leqslant i \leqslant k$ satisfy Eq. (11) with equality. Also, the k solutions ($j \in \{1,\dots,k\}$) in which b_{v_i} is selected for $1 \leqslant i \leqslant j$ and a_{v_i} for $j+1 \leqslant i \leqslant k$ satisfy Eq. (11) with equality. These $2k = \dim X(\text{PCSP})$ solutions are affinely independent. □

4.2. The clique-cycle inequalities

A second class of facet defining valid inequalities are the *clique-cycle inequalities*. Let the induced

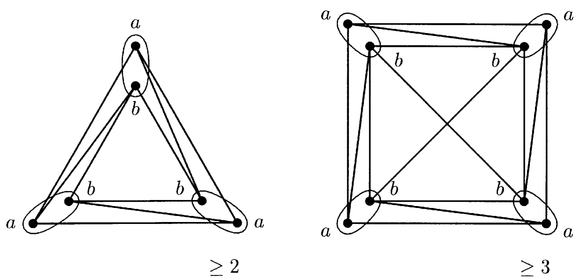

Fig. 4. Clique-cycle inequality.

subgraph $G_S=(S,E_S)$ be a k-clique, then a k-clique-cycle inequality, $k\geqslant 3$, is defined by

$$\sum_{i=1}^{k} z(v_i,A_{v_i},v_{i+1},D_{v_{i+1}}) + \sum_{i<j} z(v_i,B_{v_i},v_j,B_{v_j}) \geqslant k-1 \qquad (12)$$

with $k+1\equiv 1$. Fig. 4 shows clique-cycle inequalities for $k=3$ and 4.

It should be noted that for a subset of three vertices of the constraint graph the clique-cycle inequality and the cycle inequality describe the same facet.

Theorem 4.2. *The k-clique-cycle inequalities, $k\geqslant 3$, are valid and facet defining for X(PCSP).*

Proof. By the results of Section 3 it is sufficient to prove that the k-clique-cycle inequalities are facet defining for X(PCSP) defined by the k-clique constraint graph and $A_{v_i}=\{a_{v_i}\}$, $B_{v_i}=\{b_{v_i}\}$, $i=1,\ldots,k$.

Consider an arbitrary solution x. Whenever a_{v_i} is selected for some i, then the edge $\{v_i,v_{i+1}\}$ (or $\{v_k,v_1\}$ whenever $i=k$) contributes exactly one to the left-hand side of Eq. (12), independent of the element selected for v_{i+1}. If both b_{v_i} and b_{v_j} are selected, then the edge $\{v_i,v_j\}$ contributes exactly one to the left-hand side of Eq. (12). Hence, if b_v is selected for p vertices (and consequently a_v is selected for $k-p$ vertices), the total contribution to the left-hand side is $\binom{p}{2}+(k-p)\geqslant k-1$ for all integer p.

A clique-cycle inequality is satisfied with equality, if b_v is selected for either 1 or 2 vertices. These $\binom{k}{1}+\binom{k}{2}=k+k(k-1)/2=\dim X(\text{PCSP})$ solutions are affinely independent. □

5. Computational results

A first test of the quality of the valid inequalities described above is done on 11 instances with $|D_v|=2$ for $v\in V$. These instances are subproblems of the `celar8` instance of the CALMA-project. In the Combinatorial Algorithms for Military Applications (CALMA)-project researchers from England, France, and the Netherlands tested different combinatorial algorithms on the same set of frequency assignment problems. Results of the CALMA-project as well as all test problems are available by anonymous ftp from `ftp.win.tue.nl` in the directory `/pub/techreports/CALMA`. For these frequency assignment problems, Kolen [3] described a genetic algorithm in which the crossover is optimized, i.e. given two solutions (the parents) we would like to obtain the best-possible solution among all solutions that can be generated with the parents. So the crossover problem corresponds to a PCSP with at most two values per domain. By applying the cycle and clique-cycle inequalities these subproblems can be solved efficiently. To illustrate the efficiency of the classes of inequalities, we have selected the already mentioned 11 subproblems. We used the callable library of CPLEX 4.0 to solve the linear programming relaxation (v_{LP}), the (0,1)-programming problem (v_{IP}) as well as the linear programming relaxation with 3-cycle valid inequalities (v_3). The separation of violated valid inequalities was done by enumeration of all valid inequalities with $k=3$ (i.e. four valid inequalities for each 3-cycle were available). For all instances we have $|V|=458$ and $|E|=1655$. The results are presented in Table 1. The program written in C++ was running on a DEC 2100 A500MP workstation with 128Mb internal memory. Table 1 shows that for all instances the LP-relaxation with 3-cycle valid inequalities gives an integer solution. The number of violated inequalities which had to be added is given in the last column. The computation times were on average reduced by 76.4%.

An instance with a large gap between LP and IP is `p1`. This instance has 708 vertices and 1677 edges

Table 1
Computational results $|D_v| = 2$

Instance	v_{LP}	v_3	v_{IP}	CPU v_{LP}	CPU v_3	CPU v_{3+IP}	CPU v_{IP}	#v.i.
c8_1	848.5	986	986	8.8	18.1	18.1	78.0	1104
c8_2	721	836	836	8.7	11.4	11.4	48.4	497
c8_3	630.5	747	747	7.8	13.1	13.1	63.1	771
c8_4	802	834	834	8.0	10.9	10.9	35.4	1243
c8_5	627.5	729	729	7.5	11.3	11.3	35.7	608
c8_6	695	717	717	8.6	12.0	12.0	31.5	907
c8_7	836	894	894	8.2	9.9	9.9	39.1	267
c8_8	757	835	835	7.2	10.5	10.5	71.2	747
c8_9	769	866	866	9.2	12.6	12.6	54.9	610
c8_10	768.5	812	812	8.1	10.0	10.0	37.7	215
c8_11	622	814	814	7.3	16.0	16.0	187.1	1259
p1	35.5	104.5	110	6.6	25.5	152.4	–	266

Table 2
Computational results $|D_v| \geq 2$

Instance	$\|D_v\|$	Gap closed by		CPU-time for				
		3-cycle	3+4-cycle	v_{LP}	v_3	v_{3+4}	v_{3+IP}	$v_{(3+4)+IP}$
celar6a	2	99.9	99.9	0.7	3.1	3.1	3.1	3.5
celar6b	3	98.8	99.2	1.6	40.8	47.0	117.3	84.1
celar6c	4	89.3	92.0	3.1	327.0	471.7	36067.3	11785.5
celar6d	5	97.3	97.0	3.9	11510.9	12516.1	19647.9	22501.2
celar6e	6	95.6	97.0	4.5	50380.7	53793.1	208617.0	75570.5

(again all domains contain two values). The 3-cycle inequalities close 92.6% the gap between LP and IP. With these valid inequalities CPLEX needed 113 branch-and-bound nodes to obtain and prove the optimal value. CPLEX was not able to solve this instance to optimality without adding valid inequalities.

We also tested the cycle-inequalities on some instances with more than two elements per domain. Table 2 reports the results for five instances with 100 vertices, 350 edges, and 2, 3, 4, 5, or 6 elements in each domain. These instances are obtained by arbitrarily selecting a subset of the domain elements from the celar6 instance of the CALMA-project. Given the 3-cycles and 4-cycles in the graph, the separation of a violated inequality for each cycle was done in a heuristic way. If no violated inequalities were found, we started the branch-and-bound procedure of CPLEX. Table 2 shows for each instance the percentage of the gap between LP and IP that is closed in the case we only separate 3-cycle inequalities and in the case we separate both 3-cycle and 4-cycle inequalities. Also the cpu-times for LP, 3-cycle separation, 3-cycle and 4-cycle separation, IP obtained by 3-cycle separation, and IP obtained by 3-cycle and 4-cycle separation are reported. For all instances on average 96% of the gap between LP and IP is closed with the 3-cycle inequalities, whereas on average 97% of the gap is closed with the 3-cycle and 4-cycle inequalities. Moreover, the total computation time with the 4-cycle inequalities is substantially reduced for most instances.

6. Concluding remarks

In this paper we introduced the cycle-inequalities and clique-cycle inequalities for the PCSP. For instances with small domains the 3-cycle and 4-cycle inequalities close the gap between LP and IP substantially. For the separation of these inequalities we used a simple heuristic. Given these results future work will be done on the complexity of the separation problem for cycle inequalities and clique-cycle inequalities as well as the implementation of these classes of valid inequalities in a branch-and-cut framework (separation in a exact and/or heuristic way). In a future paper we also hope to report solutions for large-size real-life PCSPs like the complete CALMA-instances where each domain consists of 40 or 50 elements.

References

[1] K.I. Aardal, A. Hipolito, C.P.M. van Hoesel, B. Janssen, A branch-and-cut algorithm for the frequency assignment problem, Research Memorandum 96/011, Maastricht University, 1996.

[2] M. Garey, D. Johnson, L. Stockmeyer, Some simplified NP-complete graph problems, Theoret. Comput. Sci. 1 (1976) 237–267.

[3] A.W.J. Kolen, A genetic algorithm for the partial binary constraint satisfaction problem: an application to a frequency assignment problem, Working paper, 1997.

[4] R. Müller, A.S. Schulz. Transitive packing, in: W.H. Cunningham, S.T. McCormick, M. Queyranne (Eds.), Integer Programming and Combinatorial Optimization, pp. 430–444, Vancouver, British Columbia, Canada, 1996. 5th Int. IPCO Conf.

[5] G.L. Nemhauser, L.A. Wolsey,Integer and Combinatorial Optimization, Wiley, New York, 1988.

[6] M. Padberg,The boolean quadric polytope: some characteristics, facets and relatives, Math. Programming 45 (1989) 139–172.