On the system of two all_different predicates

G. Appa[1], D. Magos[2], I. Mourtos[1]

[1]London School of Economics and Political Science
Houghton Street, London WC2A 2AE

[2]Technological Educational Institute of Athens
Ag. Spyridonos Str., Egaleo 122 10, Greece

Working Paper No: LSEOR 05.74 ISBN: 07530 1696 6

First published in Great Britain in 2005
by the Department of Operational Research
London School of Economics and Political Science

Typeset, printed and bound by:
The London School of Economics and Political Science
Houghton Street
London WC2A 2AE

Working Paper No: LSEOR 05.74 ISBN No: 07530 1696 6

On the system of two all_different predicates

G. Appa[1], D. Magos[2,*], I. Mourtos[1]

[1]London School of Economics, London WC2A 2AE, UK.

email:{g.appa,j.mourtos}@lse.ac.uk

[2]Technological Educational Institute of Athens, Ag. Spyridonos Str., Egaleo 122 10, Greece.

email:dmagos@teiath.gr

Keywords: combinatorial problems, all_different predicate, constraint logic programming, facet

1 Introduction

Numerous real-life problems require certain variables to be assigned different values. This requirement is encapsulated in *constraints of difference.* If x_1, x_2 denote two problem variables, the (nonlinear) constraint of difference is $x_1 \neq x_2$. Given that variables x_1,..., x_n must all be pairwise different, a constraint of the type *all_different*$(x_1, ..., x_n)$ can be used to formulate in a compact manner the $\frac{n(n-1)}{2}$ binary constraints of difference. Such an n-ary constraint is also called a *predicate* because it imposes a logical condition on its variable set. *Constraint Programming (CP)* makes use of such elaborate predicates in order to provide a natural modelling framework ([2]). Such models are solved by CP techniques designed to produce feasible solutions. Alternatively, *Integer Programming (IP)* methods can be employed in cases where a logic predicate can be represented by linear inequalities involving integer variables ([1]). Apparently, such representations are important not only because they enrich the arsenal of resolution techniques but also because they motivate the integration of CP and IP in a unified modelling and algorithmic framework (see [3]).

An efficient representation of a predicate must be *tight*, i.e. it must include facet-defining inequalities of the convex hull of integer solutions satisfying the predicate. Such representations have been proposed for the *all_different* predicate ([8]), for *cardinality rules* ([6]) and for the *sum* constraint ([7]). A next step would be to derive such representations for sets of more that one predicates. The current paper works towards this direction by studying a system of two *all_different* constraints which may share a number of variables. In particular, we examine the polytope defined by the convex hull of integer vectors satisfying the system of the two *all_different* predicates. The dimension of this polytope is established and subsequently two classes of facet-defining inequalities are exhibited. These classes are of exponential size, a fact that

*Corresponding address: D. MAGOS, 30 Theodorou Geometrou Str., Athens 11743, Greece. Email:dmagos@teiath.gr

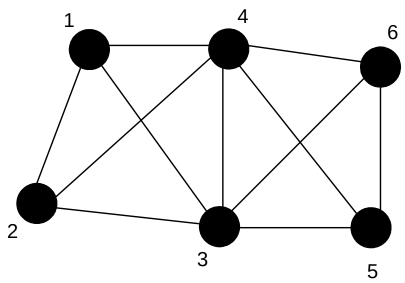

Figure 1: A graph-coloring example

prohibits their explicit use in a *Linear Programming (LP)* model. We resolve this difficulty by introducing a separation algorithm of low complexity, which provides only the facet-defining inequalities violated by a given vector. We also note that all these results can be directly applied to the *optimization* problem involving the (min-)maximization of a linear function over the system of the two *all_ different* predicates.

2 Mathematical formulation and applications

The system consists of two *all_ different* predicates, each including n variables. Let J_1 (J_2) denote the set indexing the variables of the first (second) predicate, where $|J_1| = |J_2| = n$. Let also D ($D \subset \mathbb{Z}$) denote the domain of each variable with $|D| = k$. For the system to be feasible, it must be that $k \geq n$. For simplicity, assume $D = \{0, 1, ..., k-1\}$. The system can be written as follows.

$$\textit{all\_ different}\{x_j : j \in J_1\}, \tag{1}$$

$$\textit{all\_ different}\{x_j : j \in J_2\}, \tag{2}$$

$$x_j \in D, \forall j \in J_1 \cup J_2$$

Let T denote the (possibly empty) subset of indices of the variables appearing in both predicates. In short, $T = J_1 \cap J_2$ with $|T| = t$. We denote as $I_p = J_p \backslash T$, for $p = 1, 2$, i.e. I_p is the index set of the non-common variables of each predicate. Let us provide two examples.

Example 1 *(Graph coloring [4]) Consider the graph of Figure 1. We wish to color each node in such a way that the endpoints of every edge are assigned a different color. For simplicity, assume that there are four colors available for every node, namely* $D = \{red, blue, green, orange\}$*. Let* x_i *denote the color used for node* i*, with* $D_i = D$*, for* $i = 1, ..., 6$*. Observe that this graph is formed by two cliques that have two nodes in common. Clearly, the colors assigned to the nodes of each clique must be pairwise different. Therefore, the* *coloring problem for this graph can be modelled via two all_ different predicates:*

$$all\_\ different\{x_1, x_2, x_3, x_4\},$$
$$all\_\ different\{x_3, x_4, x_5, x_6\}$$

According to our notation, $J_1 = \{1, 2, 3, 4\}$, $J_2 = \{3, 4, 5, 6\}$, $T = \{3, 4\}$.

Example 2 *(Timetabling) Consider a two-day course, where students are allocated into five groups* $\{a, b, c, d, e\}$. *Every group is assigned to a single teacher on each day of the course. To minimise the effort for teachers working on both days, the timetable should assign to them a single group throughout the course. Let the teachers available on the first and second day of the course be* $\{1, 2, 3, 4, 5\}$ *and* $\{3, 4, 5, 6, 7\}$, *respectively, i.e. teachers* $\{3, 4, 5\}$ *are teaching on both days. If* x_i *denotes the group tutored by teacher* i, *with* $D_i = \{a, b, c, d, e\}$, *the timetabling problem for this course is modelled as follows:*

$$all\_\ different\{x_1, x_2, x_3, x_4, x_5\},$$
$$all\_\ different\{x_3, x_4, x_5, x_6, x_7\}$$

Again, our notation implies that $J_1 = \{1, 2, 3, 4, 5\}$, $T = \{3, 4, 5\}$ *and* $I_1 = \{1, 2\}$.

The convex hull of integer solutions to the system (1), (2) is denoted as P_I, i.e.

$$P_I = \text{ conv}\{x \in D^{2n-t} : (1), (2) \text{ are satisfied}\}$$

Let P_L denote a linear programming (LP) relaxation of P_I. For $k > n$, we consider P_L to be

$$k - 1 \geq x_j \geq 0, \forall j \in J_1 \cup J_2$$

whereas if $k = n$, P_L is described by

$$\sum\{x_j : j \in J_1\} = \frac{n(n-1)}{2}, \tag{3}$$
$$\sum\{x_j : j \in I_2\} - \sum\{x_j : j \in I_1\} = 0, \tag{4}$$
$$x_j \geq 0, \forall j \in J_1 \cup J_2$$

To facilitate the discussion of the following section, we adopt some conventions. For a matrix X, let $X(i, j)$ denote the element appearing at row i and column j. Let $\#\operatorname{col}(X)$ ($\#\operatorname{row}(X)$) denote the number of columns (rows) of X. Assume that Y is another matrix such that $\#\operatorname{col}(Y) \leq \#\operatorname{col}(X)$ and $\#\operatorname{row}(Y) \leq \#\operatorname{row}(X)$.

For conciseness, we introduce the operation $Y \leftarrow X$ to imply the assignment of $X(i,j)$ to $Y(i,j)$, where indices i and j span only the rows and columns of matrix Y.

3 Facets of P_I

We commence the polyhedral analysis of P_I by establishing its dimension. By definition, $P_I \subset P_L$, therefore $\dim P_I \leq \dim P_L$. We will prove that $\dim P_I = \dim P_L$ by exhibiting $\dim P_L + 1$ *affinely* independent vectors of P_I. First observe that, for $k > n$, P_L is full-dimensional, whereas for $k = n$ P_L is defined in terms the equality constraints (3), (4), which form a system of full row rank.

Theorem 3

$$\dim P_I = \begin{cases} 2n - t, & \text{if } k > n, \\ 2(n-1) - t, & \text{if } k = n \end{cases}$$

Proof. Let $T = \{n-t+1, ..., n\}$, $I_1 = J_1 \setminus T = \{1, ..., n-t\}$, $I_2 = J_2 \setminus T = \{n+1, ..., 2n-t\}$. Consider a matrix B, where each row defines an integer point of P_I and each column is associated with a specific variable. Thus, entry $B(i,j)$ is the value of variable x_j at point $i \in P_I$. For $k = n$, we consider $\#\,\text{row}(B) = 2n - t - 1$, whereas for $k > n$ $\#\,\text{row}(B) = 2n - t$.

Let the first $n-t$ columns correspond to variables $x_1, ..., x_{n-t}$, the next t columns to variables $x_{n-t+1}, ..., x_n$ and the last $n-t$ columns to variables $x_{n+1}, ..., x_{2n-t}$. Hence, the sets of indices I_1, T, I_2 partition the columns of B into three sets. By also splitting the rows of B into two sets, we impose the following partitioning of B into six submatrices:

$$B = \begin{bmatrix} C_{I_1} & C_T & C_{I_2} \\ D_{I_1} & D_T & D_{I_2} \end{bmatrix}$$

Clearly, $\#\,\text{col}(C_{I_1}) = \#\,\text{col}(D_{I_1}) = n - t$, $\#\,\text{col}(C_T) = \#\,\text{col}(D_T) = t$ and $\#\,\text{col}(C_{I_2}) = \#\,\text{col}(D_{I_2}) = n - t$. To illustrate the contents of B, we initially examine the first three submatrices. Matrices C_{I_1}, C_T and C_{I_2} include n rows. The submatrix formed by C_{I_1} and C_T contains all cyclic permutations of elements $\{n-1, 0, 1, ..., n-2\}$. It is illustrated next (the vertical line separates the entries of C_{I_1} from these of C_T)

$$[C_{I_1}|C_T] = \left[\begin{array}{cccccc|ccc} \text{n-1} & 0 & \cdots & \text{n-t-4} & \text{n-t-3} & \text{n-t-2} & \text{n-t-1} & \cdots & \text{n-2} \\ \text{n-2} & \text{n-1} & \cdots & \text{n-t-5} & \text{n-t-4} & \text{n-t-3} & \text{n-t-2} & \cdots & \text{n-3} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 1 & \cdots & \text{n-t-3} & \text{n-t-2} & \text{n-t-1} & \text{n-t} & \cdots & \text{n-1} \end{array}\right]$$

Also $C_{I_2} \leftarrow C_{I_1}$, i.e. C_{I_2} is identical to C_{I_1}.

The number of rows for D_{I_1}, D_T, D_{I_2} depends on k: for $k = n$ it equals $n - t - 1$, whereas for $k > n$ it equals $n - t$. These two cases are examined separately in our proof. We set $D_{I_1} \leftarrow C_{I_1}$ to emphasise that matrix D_{I_1} receives only the first $\#\,\text{row}(D_{I_1})$ rows of matrix C_{I_1}. For example, in the case of $k = n$, the last $t+1$ rows of $C_{I_!}$ are not contained in D_{I_1}.

$$D_{I_1} = \begin{bmatrix} \text{n-1} & 0 & \cdots & \text{n-t-4} & \text{n-t-3} & \text{n-t-2} \\ \text{n-2} & \text{n-1} & \cdots & \text{n-t-5} & \text{n-t-4} & \text{n-t-3} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \text{t+1} & \text{t+2} & \cdots & \cdots & \cdots & \cdots \end{bmatrix}$$

In an analogous manner, we set $D_T \leftarrow C_T$.

To better illustrate the construction of submatrix D_{I_2}, it is convenient to consider an alternative partitioning of B:

$$B = \begin{bmatrix} C_{I_1,T} & \bar{C}_{I_2} & p \\ D_{I_1,T} & \bar{D}_{I_2} & q \end{bmatrix}$$

where $C_{I_1,T} = [C_{I_1}|C_T]$, $D_{I_1,T} = [D_{I_1}|D_T]$, $C_{I_2} = [\bar{C}_{I_2}|p]$, $D_{I_2} = [\bar{D}_{I_2}|q]$ with p, q being column vectors. The actual contents of $\bar{D}_{I_2}$ and q are analysed within the following cases.

Case 1 $k = n$

Matrix D_{I_1} includes two top-left to bottom-right diagonals, namely d_1, d_2, of maximum size (i.e. $n-t-1$). Diagonals d_1 and d_2 include the values $B(n+r, r)$ and $B(n+r, r+1)$, respectively, for $r = 1, ..., n-t-1$. We denote as S the triangular part of D_{I_1} including all the elements above d_1. Observe that diagonal d_2 is included in S, while diagonal d_1 is not. Thus, S includes the elements $B(n+i, j)$, for $i = 1, ..., n-t-1$, $j = i+1, ..., n-t$. Also let Q denote the triangular part of D_{I_1} including all the elements below d_1 (but not d_1 itself), i.e. Q includes the elements $B(n+i, j)$, for $i = 2, ..., n-t-1$, $j = 1, ..., i-1$. Matrix $\bar{D}_{I_2}$ is identical to D_{I_1} except from its diagonal d_1, which is omitted and instead placed in column vector q. Hence, matrix $\bar{D}_{I_2}$ contains d_2 as its single main diagonal. A schematic illustration of D_{I_1} and D_{I_2} is shown below.

$$D_{I_1} = \begin{bmatrix} \ddots & & S \\ & d_1 & \\ Q & & \ddots \end{bmatrix}, \qquad D_{I_2} = \left[\begin{array}{cc|c} \ddots & S & \\ Q & \ddots & d_1 \end{array}\right]$$

As an example, we illustrate matrix B for $n = 6$ and $t = 2$. Notice that the bordered elements belong to diagonal d_1 and appear also at the last column of D_{I_2}.

$$B = \left[\begin{array}{cccc||cc||cccc}
5 & 0 & 1 & 2 & 3 & 4 & 5 & 0 & 1 & 2 \\
4 & 5 & 0 & 1 & 2 & 3 & 4 & 5 & 0 & 1 \\
3 & 4 & 5 & 0 & 1 & 2 & 3 & 4 & 5 & 0 \\
2 & 3 & 4 & 5 & 0 & 1 & 2 & 3 & 4 & 5 \\
1 & 2 & 3 & 4 & 5 & 0 & 1 & 2 & 3 & 4 \\
0 & 1 & 2 & 3 & 4 & 5 & 0 & 1 & 2 & 3 \\
\hline\hline
\boxed{5} & 0 & 1 & 2 & 3 & 4 & 0 & 1 & 2 & 5 \\
4 & \boxed{5} & 0 & 1 & 2 & 3 & 4 & 0 & 1 & 5 \\
3 & 4 & \boxed{5} & 0 & 1 & 2 & 3 & 4 & 0 & 5
\end{array}\right]$$

The following series of operations amount to subtracting the columns of $[C_{I_1}|D_{I_1}]^T$ from those of $[C_{I_2}|D_{I_2}]^T$. We set $O \leftarrow C_{I_2} - C_{I_1}$ and $E \leftarrow D_{I_2} - D_{I_1}$. Observe that O is a matrix of zeros since $C_{I_1} = C_{I_2}$. By substituting in B the submatrix $[C_{I_2}|D_{I_2}]^T$ by $[O|E]^T$, we obtain matrix $\bar{B}$. Evidently, $\operatorname{rank} B = \operatorname{rank} \bar{B}$. Let $E = [\bar{E}|\bar{q}]$ and $O = [\bar{O}|0]$, where $\bar{q}$ is the last column of E and 0 is a column vector of zeros. Analytically,

$$\bar{B} = \begin{bmatrix} C_{I_1,T} & \bar{O} & 0 \\ D_{I_1,T} & \bar{E} & \bar{q} \end{bmatrix}$$

By construction, $\bar{E}$ is a square upper triangular matrix with non-zero elements in its main diagonal. To see this, notice that Q appears in the same position both in D_{I_1} and D_{I_2} and d_1, d_2 differ in all elements. As a result, $\bar{E}(i,i) = d_2(i) - d_1(i) \neq 0$, for every i. Hence, $\bar{E}$ is non-singular. $C_{I_1,T}$ is a non-negative square cyclic matrix, therefore also non-singular. The determinant of the matrix obtained from $\bar{B}$ by deleting its last column (i.e. $[0|\bar{q}]^T$) is equal to $\det C_{I_1,T} \cdot \det \bar{E} \neq 0$. Hence, the matrix $\bar{B}$ (B also) is of full row rank. Thus, the rows of B illustrate $2n - t - 1$ linearly independent vectors of P_I.

Case 2 $k > n$

For this case we must exhibit $2n - t + 1$ affinely independent points of P_I. Matrix B remains as in the previous case except for a few changes. First, recall that submatrix D_{I_1} (and also D_T, D_{I_2}) includes one additional row in this case. As a result, D_{I_1} is now a square submatrix with a single diagonal of maximum size (i.e. $n - t$), namely d_1. Moreover, the last entry of d is set to n, i.e. $B(2n - t, 2n - t) \leftarrow n$. Consider the matrix

$$B' = \begin{bmatrix} B \\ g \end{bmatrix}$$

where row $g = (0, 1, ..., n-2, n, 0, ..., n-t-1)$. It is sufficient to prove that the $2n-t+1$ rows of B', each corresponding to an integer point of P_I, are affinely independent. Based on the definition of affine independence ([5]), this is equivalent to showing the non-singularity of the matrix

$$M = [B' | \, e] = \left[\begin{matrix} B \\ g \end{matrix} \middle| \, e \right]$$

where e ($\#\text{row}(e) = 2n-t+1$) is a column vector of 1s. With respect to matrix B', we subtract the first $n-t$ columns from the last $n-t$ columns (i.e., perform the same elementary column operations as in the case of $k = n$), thus obtaining the submatrix $[\bar{B}|\bar{g}]^T$, where $\bar{g} = (0, 1, ..., n-2, n, 0, ..., 0)$. These operations transform the matrix M to $\bar{M}$ which has the same rank as M. We consider the following partitioning of $\bar{M}$.

$$\bar{M} = \begin{bmatrix} C_{I_1,T} & O & e_1 \\ D_{I_1,T} & E & e_2 \\ & \bar{g} & 1 \end{bmatrix}$$

where O, E are defined exactly as in the previous case and e_1, e_2 are column vectors of ones of appropriate size. Also observe that O, E are square submatrices with O containing zeros and E being upper diagonal with non-zero entries in its leading diagonal.

To transform the last column of $\bar{M}$ into the vector $(0, ..., 0, 1)^T$, we perform the following elementary operations: we subtract (a) the last row of $[C_{I_1,T}|O|e_1]$ from all other rows of this submatrix, and, (b) the row $[\bar{g}|1]$ from the last row of $[C_{I_1,T}|O|e_1]$ and from the rows of $[D_{I_1,T}|E|e_2]$. The derived matrix, namely M', has the same rank as $\bar{M}$. Analytically,

$$M' = \begin{bmatrix} C'_{I_1,T} & O' \\ D'_{I_1,T} & E' \end{bmatrix}, \quad \text{where} \quad E' = \begin{bmatrix} E & 0 \\ 0^T & 1 \end{bmatrix}, C'_{I_1,T} = \begin{bmatrix} \text{n-1} & \text{-1} & \text{-1} & \text{-1} & \cdots & \text{-1} & \text{-1} & \text{-1} \\ \text{n-2} & \text{n-2} & \text{-2} & \text{-2} & \cdots & \text{-2} & \text{-2} & \text{-2} \\ \text{n-3} & \text{n-3} & \text{n-3} & \text{-3} & \cdots & \text{-3} & \text{-3} & \text{-3} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 2 & 2 & 2 & 2 & \cdots & 2 & \text{-(n-2)} & \text{-(n-2)} \\ 1 & 1 & 1 & 1 & \cdots & 1 & 1 & \text{-(n-1)} \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & \text{-1} \end{bmatrix}$$

and $O' = [O|0]$ with 0 being a zero column vector.

It is easy to see that E' is non-singular. To show that $C'_{I_1,T}$ is also non-singular, we perform operations in terms of rows and columns of M' that leave both O' and E' unaffected. First, we permute the columns of $C'_{I_1,T}$ in such a way that its first column is shifted between columns $n-1$ and n. Then, to each row $j \in \{2, ..., n-1\}$, we add the multiple of the first row by $(n-j)$. The resulting matrix is upper diagonal with non zero elements in its leading diagonal. Hence, $\det M' = \det C'_{I_1,T} \cdot \det E' \neq 0$ implying that $\bar{M}$, M

are non-singular. ■

The facets of the polytope defined as the convex hull of all vectors satisfying a single *all_ different* constraint are given in [8]. For the system studied here, the corresponding classes of inequalities are

$$\sum\{x_s \quad : \quad s \in S\} \geq \frac{|S|\,(|S|-1)}{2}, \forall S \subseteq J_p, p = 1,2, \tag{5}$$

$$\sum\{x_s \quad : \quad s \in S\} \leq \frac{|S|\,(2k-|S|-1)}{2}, \forall S \subseteq J_p, p = 1,2 \tag{6}$$

We note that if $k = n$ then $S \subseteq J_p$ is replaced by $S \subset J_p$ (strict inclusion). In this case, (5) and (6), taken for $S = J_p$, are satisfied as equalities by all points of P_I therefore they cannot be facet-defining. For all other cases, we prove that (5), (6) induce facets of P_I.

We illustrate the result for inequality (5) and for $S \subset J_2$, all other cases of (5), (6) being symmetrical. Hence, for $H = \{i_1, i_2, ..., i_h\} \subset J_2$, consider the inequality

$$x_{i_1} + x_{i_2} + \cdots + x_{i_h} \geq \frac{h(h-1)}{2} \tag{7}$$

Define $F = \{x \in P_I : x_{i_1} + x_{i_2} + \cdots + x_{i_h} = \frac{h(h-1)}{2}\}$. Let A denote the coefficient matrix of the minimum equality system for P_I. To prove that (7) is facet-defining, we show that if there exists an equation $ax = a_0$ satisfied by all points of F, then $[a|a_0]$ can be written as a linear combination of the rows of A and the coefficients of (7) (see [5]). Notice that, for $k = n$, the minimum equality system is defined by the linearly independent equalities (3), (4), while there exists no equality system for $k > n$. Therefore, rank $A = 2$ for $k = n$, while rank $A = 0$ for $k > n$. We examine these two cases separately.

Theorem 3 *For $k = n$, $a \in \mathbb{R}^{2n-t}$, $a_0 \in \mathbb{R}$, if $ay = a_0$ holds for every $y \in F$, there exist scalars $\lambda_1, \lambda_2, \pi$ such that*

$$a_i = \begin{cases} \lambda_1,\ i \in T \setminus H, \\ \lambda_2,\ i \in I_2 \setminus H, \\ \lambda_1 - \lambda_2,\ i \in I_1, \\ \lambda_1 + \pi,\ i \in T \cap H, \\ \lambda_2 + \pi,\ i \in I_2 \cap H \end{cases} \tag{8}$$

and

$$a_0 = \lambda_1 \frac{n(n-1)}{2} + \pi \frac{h(h-1)}{2} \tag{9}$$

Proof. Evidently, there exists at least one $s \in (I_2 \cup T) \backslash H$. If $s \in I_2$ define $\lambda_1 = a_1 + a_s$, $\lambda_2 = a_s$, whereas if $s \in T$ define $\lambda_1 = a_s$, $\lambda_2 = a_s - a_1$. By substituting in (7), we obtain the coefficients of the a

Table 1: Values of a_i

	$s \in I_2$	$s \in T$
$i \in T \setminus H$	$a_1 + a_s$	a_s
$i \in I_2 \setminus H$	a_s	$a_s - a_1$
$i \in I_1$	a_1	
$i \in T \cap H$	$a_1 + a_s + \pi$	$a_s + \pi$
$i \in I_2 \cap H$	$a_s + \pi$	$a_s - a_1 + \pi$

Table 2: Proving (8) for $i \notin H$

	$s \in I_2$	$s \in T$
$i \in T \setminus H$	$x_1 = 1, x_s = 1, x_i = 0,$ $\bar{x}_1 = 0, \bar{x}_s = 0, \bar{x}_i = 1$	$x_i = 0, x_s = 1,$ $\bar{x}_i = 1, \bar{x}_s = 0$
$i \in I_2 \setminus H$	$x_s = 0, x_i = 1,$ $\bar{x}_i = 0, \bar{x}_s = 1$	$x_1 = 1, x_i = 1, x_s = 0,$ $\bar{x}_1 = 0, \bar{x}_i = 0, \bar{x}_s = 1$
$i \in I_1$	$x_1 = 0, x_i = 1,\ \bar{x}_1 = 1, \bar{x}_i = 0$	

vector $(a_i, \forall i \in J_1 \cup J_2)$ illustrated in Table 1. We must prove that each a_i is equal to the value depicted in the corresponding cell of the table. We briefly describe the methodology followed. For each case, we consider two integer points $x, \bar{x} \in F$. By hypothesis, both points satisfy $ay = a_0$, therefore equation $ax = a\bar{x}$ holds. By properly selecting x and $\bar{x}$, the desired result is obtained after cancelling identical terms in equation $ax = a\bar{x}$.

We illustrate analytically this approach for $i \in T \setminus H$, and for $s \in I_2$. Assume an integer point $x \in F$ such that $x_1 = 1, x_s = 1, x_i = 0$. Notice that this point satisfies both *all_ different* constraints and also (7) as an equality. By hypothesis, it holds that $ax = a_0$ or analytically:

$$a_1 + a_s + 0 \cdot a_i + \sum \{a_j x_j : j \in (I_1 \cup I_2 \cup T) \backslash \{1, s, i\}\} = a_0 \tag{10}$$

Also consider the integer point $\bar{x}$, such that $\bar{x}_1 = 0, \bar{x}_s = 0, \bar{x}_i = 1, \bar{x}_j = x_j$ for all $j \in (I_1 \cup I_2 \cup T) \backslash \{1, s, i\}$. Observe that $\bar{x} \in F$. We can write equation $a\bar{x} = a_0$ in the following form:

$$0 \cdot a_1 + 0 \cdot a_s + a_i + \sum \{a_j x_j : j \in (I_1 \cup I_2 \cup T) \backslash \{1, s, i\}\} = a_0 \tag{11}$$

It is easy to see that equations (10) and (11) imply $a_i = a_1 + a_s$, as required.

The remaining cases for $i \notin H$ can be checked in the same fashion through Table 2. This table depicts only the relevant values of $x, \bar{x} \in F$; the remaining ones have identical values in both x and $\bar{x}$, thus resulting in the corresponding terms of $ax = a\bar{x}$ to cancel out.

It remains to prove our claim for $i \in H$. For $s \in I_2$, define

$$\pi_i = \begin{cases} a_i - a_1 - a_s, & \text{for } i \in T \cap H \\ a_i - a_s, & \text{for } i \in I_2 \cap H \end{cases} \tag{12}$$

We will show that all π_i are equal. Let $i_s, i_q \in T \cap H$. Consider an integer point $x \in F$ with $x_{i_q} = 1, x_{i_s} = 0$ and $\bar{x}$ having $\bar{x}_{i_s} = 1, \bar{x}_{i_q} = 0$, $\bar{x}_m = x_m$, for all $m \in (J_1 \cup J_2) \setminus \{i_s, i_q\}$. Then, equation $ax = a\bar{x}$, after cancelling identical terms and substituting the remaining terms from (12), yields $\pi_{i_s} = \pi_{i_q}$. A similar result is valid for $i_s, i_q \in I_2 \cap H$. Finally, let $i_q \in T \cap H$ and $i_s \in I_2 \cap H$. Consider an integer point $x \in F$ with $x_{i_q} = 1, x_1 = x_{i_s} = 0$ and $\bar{x}$ having $\bar{x}_1 = \bar{x}_{i_s} = 1, \bar{x}_{i_q} = 0$, $\bar{x}_m = x_m$, for all $m \in (J_1 \cup J_2) \setminus \{1, i_s, i_q\}$. Thus, $ax = a\bar{x}$ yields $a_{i_q} = a_{i_s} + a_1$. By substituting terms a_{i_q}, a_{i_s} from (12) and cancelling out identical terms, we obtain $\pi_{i_s} = \pi_{i_q} = \pi$.

For $s \in T$, define

$$\pi_i = \begin{cases} a_i + a_1 - a_s, & \text{for } i \in I_2 \cap H, \\ a_i - a_s, & \text{for } i \in T \cap H \end{cases}$$

The proof is carried out in a manner analogous to that of the previous case.

The proof of (8) is complete. To show (9), consider $s \in I_2$ and an arbitrary integer point $x \in F$. Then $ax = a_0$ can be written as

$$a_0 = \sum_{i \in I_1} a_i x_i + \sum_{i \in (I_2 \cup T) \setminus H} a_i x_i + \sum_{i \in H} a_i x_i$$

By substituting all terms from Table 1, we obtain

$$\begin{aligned} a_0 &= a_1 \sum_{i \in I_1} x_i + a_s \sum_{i \in I_2 \setminus H} x_i + (a_1 + a_s) \sum_{i \in T \setminus H} x_i \\ &\quad + (a_s + \pi) \sum_{i \in I_2 \cap H} x_i + (a_1 + a_s + \pi) \sum_{i \in T \cap H} x_i \\ &= a_1 \sum_{i \in I_1 \cup T} x_i + a_s \sum_{i \in I_2 \cup T} x_i + \pi \sum_{i \in H} x_i \\ &= (a_1 + a_s) \frac{n(n-1)}{2} + \pi \frac{h(h-1)}{2} \end{aligned}$$

In an analogous manner we show (9) for $s \in T$. ■

Establishing an analogous result for $k > n$ is simpler.

Theorem 4 *For $k > n$, $a \in \mathbb{R}^{2n-t}$, $a_0 \in \mathbb{R}$, if $ay = a_0$ holds for every $y \in F$, there exists a scalar π such that*

$$a_i = \begin{cases} 0, & i \in (J_1 \cup J_2) \setminus H, \\ \pi, & i \in H \end{cases} \tag{13}$$

and

$$a_0 = \pi \frac{h(h-1)}{2} \tag{14}$$

Proof. As in the previous proof, we define $\pi_i = a_i$, for all $i \in H$, and prove that all π_i are equal. Let $i_s, i_q \in H$. Since $H \subset J_2$, we must examine the following cases: (i) $i_s, i_q \in I_2 \cap H$, (ii) $i_s, i_q \in T \cap H$ and (iii) $i_s \in I_2 \cap H$, $i_q \in T \cap H$. For the first two cases, the proof proceeds exactly as in Theorem 3. For the third case, consider an integer point $x \in F$ such that $x_{i_q} = 1, x_{i_s} = 0$ and $x_i \neq 1$, for all $i \in I_1$. Such a point exists only for $k > n$, since there are enough values in $D \setminus \{1\}$ to be assigned to the variables indexed by I_1. Consider also $\bar{x} \in F$ such that $\bar{x}_{i_q} = 0$, $\bar{x}_{i_s} = 1$, $\bar{x}_i = x_i$, for all $i \in (J_1 \cup J_2) \setminus \{i_s, i_q\}$. Equation $ax = a\bar{x}$ yields $\pi_{i_q} = \pi_{i_s} = \pi$.

To show (14), consider an arbitrary integer point $x \in F$. Then,

$$ax = \sum_{i \in H} a_i x_i + \sum_{i \in (J_1 \cup J_2) \setminus H} a_i x_i = \pi \cdot \sum_{i \in H} x_i + 0 \cdot \sum_{i \in (J_1 \cup J_2) \setminus H} x_i = \pi \frac{h(h-1)}{2}$$

■

The above theorems imply the following.

Corollary 5 *For $n \geq 2$, inequalities (5) and (6) define facets of P_I.*

4 A separation algorithm

Linear programming can be employed to provide a point of P_L. Checking whether this solution violates the facet-defining inequalities (5), (6) constitutes the *separation* problem. Separation is important because violated inequalities can be added to the linear program, thus obtaining a tighter LP relaxation. In our case, solving the separation problem by a brute-force method is not efficient since the number of inequalities is exponential in n, i.e. equals $2(2^n - 1)$. Next, we present a polynomial-time separation algorithm which, given $x \in P_L$, either ends up with a violated inequality belonging to (5), (6) or proves that no such inequality exists. Comments are included in /* */.

Algorithm 6 */* Input $x \in P_L$ */*

Step 1: $v \leftarrow 0, u \leftarrow 0$;

Step 2: Sort the variable of x in ascending order, in terms of their values, deriving $\{x_{i_1}, x_{i_2}, ..., x_{i_n}\}$;

Step3:For $h = 1, ..., n$,

{

$v \leftarrow v + x_{i_h}$;

if $v < \frac{h(h-1)}{2}$ *then return; (/* Inequality* $\sum_{j=1}^{h} x_{i_j} \geq \frac{h(h-1)}{2}$ *is violated*/)*

$u \leftarrow u + x_{i_{n+1-h}}$;

if $u > \frac{h(2k-h-1)}{2}$ *then return; (/* Inequality* $\sum_{j=1}^{h} x_{i_{n+1-j}} \leq \frac{h(2k-h-1)}{2}$ *is violated*/)*

}

Proposition 7 *Algorithm 6 determines in $O(n \log_2 n)$ steps whether a facet of P_I described by (5) (6) is violated.*

Proof. Because of the ordering it holds that $\sum_{j=1}^{h} x_{i_j} \leq \sum\{x_s : s \in S\}$ for every $S \subseteq J_p, |S| = h$. Hence if there exists S ($|S| = h$) such that $\sum\{x_s : s \in S\} < \frac{h(h-1)}{2}$ then $\sum_{j=1}^{h} x_{i_j} < \frac{h(h-1)}{2}$. The case is similar for the inequalities (6). The complexity of Step 3 is $O(n)$. Thus, the most expensive operation is the sorting of the variables (Step 2), which can be accomplished easily in $O(n \log_2 n)$ time. ■

References

[1] K. Darby-Dowman, J. Little, Properties of some combinatorial optimization problems and their effect in the performance of integer programming and constraint logic programming, INFORMS J. Comput. 10 (1998) 276-286.

[2] P. van Hentenryck, Constraint Satisfaction in Logic Programming, MIT Press, Boston, MA, 1989.

[3] J. N. Hooker, Logic Based Methods for Optimization, Wiley-Inter-Science, New York, NY, 2000.

[4] C. Le Pape, Private communication, ILOG S.A., 2004.

[5] W. R. Pulleyblank, Polyhedral combinatorics, in: G. L. Nemhauser, A. H. G. Rinnooy Kan, M. J. Todd, (Eds.) Optimization, North-Holland, Amsterdam, 1989, pp. 371-446.

[6] H. Yan, J. N. Hooker, Tight Representation of Logic Constraints as Cardinality Rules, Math. Program. 85 (1999) 363-377.

[7] T. H. Yunes, On the sum constraint: relaxation and applications, in: P. van Hentenryck, (Ed.) Principles and Practice of Constraint Programming - CP2002, LNCS 2470, Springer, Berlin, 2002, pp. 80-92.

[8] H. P. Williams, H. Yan, Representations of the all_different predicate of constraint satisfaction in integer programming, INFORMS J. Comput. 13 (2001) 96-103.