Matrices with lexicographically-ordered rows

Gustavo Angulo
Pontificia Universidad Católica de Chile
gangulo@ing.puc.cl

September 6, 2018

Abstract

The lexicographic order can be used to force a collection of decision vectors to be all different, i.e., to take on different values in some coordinates. We consider the set of fixed-size matrices with bounded integer entries and rows in lexicographic order. We present a dynamic program to optimize a linear function over this set, from which we obtain a compact extended formulation for its convex hull.

1 Introduction

Consider a combinatorial problem whose variables include n-dimensional integer vectors $x_1, \ldots, x_m \in \mathbb{Z}_+^n$ that aim to represent m different options or solutions that must be taken into account simultaneously in the optimization process. That is the case, for instance, of uncertain problems having robustness or reliability requirements, where $x_1, \ldots, x_m$ represent m different contingency plans defined by n decision variables each. The all-different system $x_i \neq x_j$ for $i \neq j$ dictates that no two distinct decision vectors take on the same values, that is, the rows of the $m \times n$ matrix $x = [x_1, \ldots, x_m]^\top$ are different. We write x_i^t for the variable given by row i and column t.

When $n = 1$ and each $x_i \in \mathbb{Z}_+$ takes values on $\{0, \ldots, u\}$, a complete linear description of the convex hull of vectors $(x_1, \ldots, x_m)^\top \in \mathbb{Z}_+^m$ satisfying the all-different system is presented in [14]. In [11] and [3], these results are extended to multiple all-different systems acting on overlapping subsets of variables, under the condition that a so-called inclusion property holds.

When $n \geq 1$ and each $x_i \in \mathbb{Z}_+^n$ takes values on $\{0,1\}^n$, we obtain the all-different polytope introduced in [8] and further studied in [9]. It can be seen as a binarization of the previous case for $u = 2^n - 1$, where now each row of the $m \times n$ binary matrix x corresponds to a binary expansion with $n = \log(u+1)$ bits. Although no complete linear description or compact extended formulation is known for this polytope, linear optimization over it can be carried out in polynomial time [1].

A different way to enforce distinct binary vectors is to have them ordered with respect to the binary expansion they represent by including the constraints $\sum_{t=1}^n 2^{t-1} x_i^t + 1 \leq \sum_{t=1}^n 2^{t-1} x_j^t$ for $i < j$. This is equivalent to requiring that x_i is lexicographically smaller than x_j, denoted $x_i \prec x_j$, meaning that there exists $t \in \{1, \ldots, n\}$ such that $x_i^h = x_j^h$ for all $h < t$ and $x_i^t < x_j^t$. Such lexicographic constraints define the full orbitope $O_{m,n}$ introduced in [7]. Compact extended formulations having $\mathcal{O}(m^3 n)$ variables and inequalities can be obtained via dynamic programming [10] and the framework of branched polyhedral systems [6]. Moreover, for binary problems having a symmetry group acting on the coordinates of the decision vector, [5] studies the convex hull of solutions that are lexicographically maximal in their orbits, termed symretopes, and related polytopes, obtaining polynomial linear descriptions and extended formulations for specific cases.

The lexicographic order also applies to nonbinary vectors. Given vectors $\alpha, \beta \in \mathbb{Z}^n_+$, [4] and [2] characterize the convex hull of bounded integer vectors y satisfying $\alpha \preceq y \preceq \beta$, where $\preceq$ extends $\prec$ by allowing equality. The complete linear description has $\mathcal{O}(n)$ constraints, parametrized by maximal and minimal elements with respect to α and β.

In this work, we consider the case where the vectors $x_1, \ldots, x_m$ are given by general integer variables and are differentiated by the lexicographic order. For a positive integer u, let $L^u_{m,n}$ be the set of $m \times n$ matrices with entries in $\{0, \ldots, u\}$ and rows in increasing lexicographic order. Our main contributions are a dynamic program with polynomial complexity to optimize a linear function over $L^u_{m,n}$ and an extended formulation for its convex hull, denoted $\operatorname{conv}(L^u_{m,n})$, of polynomial size. More precisely, for $u \geq m-1$, the complexity of our dynamic program and the size of our formulation are $\mathcal{O}(m^3 n)$, matching those of the case $u = 1$, while for $1 < u < m-1$, they are $\mathcal{O}(um^3 n)$.

Remark 1. *In principle, we could binarize each variable of $x \in L^u_{m,n}$ with $k := \lceil \log(u+1) \rceil$ bits and write x as a linear transformation of a binary matrix $z \in L^1_{m,kn}$ via $x^t_i = \sum_{a=1}^{k} 2^{a-1} z_i^{(t-1)k+a}$. If $u+1$ is a power of 2, then this linear mapping is a one-to-one correspondence between $L^u_{m,n}$ and $L^1_{m,kn}$, and since $\operatorname{conv}(L^1_{m,kn}) = O_{m,kn}$, the results in [10] and [6] for the full orbitope yield an extended formulation for $\operatorname{conv}(L^u_{m,n})$ of size $\mathcal{O}(km^3 n)$. For $u \geq m-1$, this extended formulation is larger than ours by a factor of $k = \log(u+1)$. Moreover, if $u+1$ is not a power of 2, then $k > \log(u+1)$ and some elements of $L^1_{m,kn}$ do not project onto an element of $L^u_{m,n}$. For instance, if $z_m \in \{0,1\}^{kn}$ is an all-ones row, then it will project onto $x_m \in \mathbb{Z}^n$ having each component equal to $2^k - 1 > u$. Therefore, the extended formulations for the full orbitope will not work in this case.*

A key tool in our development is the polyhedral characterization of discrete dynamic programming of [12]. Suppose we have an acyclic directed hypergraph with vertices and hyperarcs representing states and decisions of a dynamic program, given by finite sets S and H, respectively. Here, each hyperarc in H is of the form (Δ, s), where the tail $\Delta \subseteq S$ is a subset of states that are combined to reach the head state $s \in S$. Boundary states have inbound hyperarcs with $\Delta = \varnothing$. Let us write the flow conservation constraints on the hypergraph as $Ay = e_0,\ y \geq 0$, where e_0 is a unit vector indexing the final global state of the dynamic program. Then, as [12] shows, the system $Ay = e_0,\ y \geq 0$ is total dual integral, meaning that for any integer vector d such that the primal problem $\min\{d^\top y : \ Ay = e_0,\ y \geq 0\}$ has finite optimal value, the dual problem $\max\{\pi_0 : \ A^\top \pi \leq d\}$ has an integer optimal solution, with π_0 the dual variable corresponding to the inflow constraint of the final state. In particular, this implies that the primal polyhedron is integral, as its right-hand side vector is integer [13, Chapter 22]. Moreover, if there exists a finite reference set I such that each $s \in S$ can be tagged with $I[s] \subseteq I$ satisfying certain conditions of consistency and disjointness, then the primal polyhedron has binary vertices only [12], which represent the decisions made throughout the dynamic program. More precisely, it is required that $I[\delta] \subseteq I[s]$ for all $(\Delta, s) \in H$ and $\delta \in \Delta$, and that $I[\delta] \cap I[\delta'] = \varnothing$ for all $(\Delta, s) \in H$ and distinct $\delta, \delta' \in \Delta$.

2 Properties of vertices

In this section we present some properties of the vertices of $\mathrm{conv}(L^u_{m,n})$ that will be useful in our development.

We say that a nondecreasing integer vector w has a split (resp. jump) if w has two consecutive entries whose values differ by exactly 1 (resp. by at least 2.)

Note that if $w \in \{0, \dots, u\}^m$ is the first column of a matrix in $L^u_{m,n}$ (and thus it is a nondecreasing vector) and has a split or jump, say $w_{l-1} < w_l$ for some $1 < l \leq m$, then the first $l-1$ rows are lexicographically smaller than the $m-l+1$ remaining rows.

Example 1. *For $m = 7$, $n = 3$, and $u = 3$, consider the matrix $x \in L^3_{7,3}$ given by*

$$x = \begin{array}{|c|cc|} \hline 0 & 2 & 1 \\ 0 & 2 & 3 \\ \hline 1 & 0 & 3 \\ 1 & 3 & 1 \\ 1 & 3 & 2 \\ \hline 3 & 1 & 3 \\ 3 & 2 & 0 \\ \hline \end{array}$$

The first column of x has a split and a jump, and divides the matrix into three submatrices having their first column with constant values. Therefore, the rows within each submatrix, even without considering their first entry, must be in lexicographic order as well.

For $1 \leq i \leq j \leq m$ and $1 \leq t \leq n$, let (i, j, t) denote the block of indices given by rows from i to j and columns from t to n. Let x^t_{ij} denote the submatrix of x given by block (i, j, t) as depicted in Figure 1.

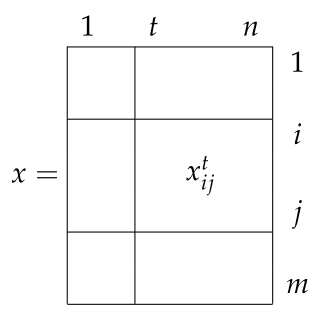

Figure 1: Block (i, j, t) defining x^t_{ij}.

Proposition 2. *Let $x \in L^u_{m,n}$ be a vertex of $\mathrm{conv}(L^u_{m,n})$ and let w be its first column. Then $w_1 = 0$ or $w_m = u$, and w has at most one jump.*

Proof. If $0 < w_1$ and $w_m < u$, then we can increase and decrease all the entries of w by 1 and obtain two distinct matrices in $L^u_{m,n}$ whose average is precisely x. Therefore $w_1 = 0$ or $w_m = u$. If w has two jumps, say $w_{i-1} + 1 < w_i$ and $w_j < w_{j+1} - 1$ with $i \leq j$, then we can increase and decrease the entries of w with indices in $\{i, \dots, j\}$ by 1 and obtain two distinct matrices in $L^u_{m,n}$ whose average is again x. Therefore w has at most one jump. $\square$

The above conditions are necessary but not sufficient to have a vertex of $\text{conv}(L^u_{m,n})$ as the following example shows.

Example 2. *For $m = n = u = 2$, let*

$$x = \begin{bmatrix} 0 & 0 \\ 1 & 2 \end{bmatrix}, \quad y = \begin{bmatrix} 0 & 0 \\ 0 & 2 \end{bmatrix}, \quad z = \begin{bmatrix} 0 & 0 \\ 2 & 2 \end{bmatrix}.$$

These three matrices belong to $L^2_{2,2}$, and although the first column of x satisfies the conditions of Proposition 2, $x = \frac{1}{2}y + \frac{1}{2}z$ holds.

Still, these conditions are enough to derive a dynamic program to optimize a linear function over $L^u_{m,n}$. First, we prove the following result depicted in Figure 2. The cases below are exhaustive but not mutually exclusive.

Proposition 3. *Let $x \in L^u_{m,n}$ be a vertex of $\text{conv}(L^u_{m,n})$ and let w be its first column. Then either (i) w is constant with all entries equal to 0, (ii) w is constant with all entries equal to u, (iii) $w_1 = 0$, $w_m > 0$, and w does not have jumps, (iv) $w_1 < u$, $w_m = u$, and w does not have jumps, or (v) $w_1 = 0$, $w_m = u$, and w has at most one jump.*

Proof. If w is constant, since $w_1 = 0$ or $w_m = u$ by Proposition 2, then all entries of w are equal to 0 or u, which gives us cases (i) and (ii). Now suppose w has a jump, say $w_{l-1} + 1 < w_l$. If $0 < w_1$ (resp. $w_m < u$), then we can increase and decrease the entries of w with indices in $\{1, \dots, l-1\}$ (resp. $\{l, \dots, m\}$) by 1 and obtain two distinct matrices in $L^u_{m,n}$ whose average is equal to x. Together with Proposition 2, this gives us case (v). Cases (iii) and (iv) are the remaining possibilities in view of Proposition 2. $\square$

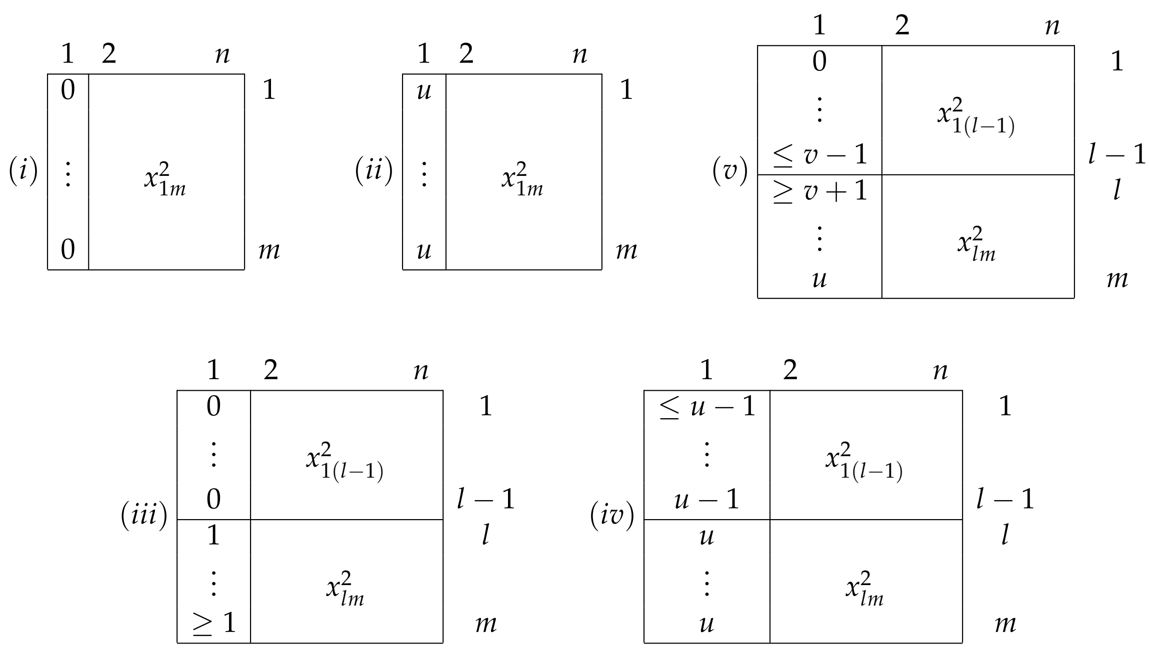

Figure 2: The five possibilities for a vertex $x = x^1_{1m}$ of $\text{conv}(L^u_{m,n})$.

3 The case $u \geq m-1$

We begin by stating the main result of this section.

Theorem 4. *The system (1)-(7) below is an extended formulation for* $\text{conv}(L^u_{m,n})$ *with* $\mathcal{O}(m^3 n)$ *variables and constraints:*

- *For* $1 \leq i \leq m$ *and* $1 \leq t \leq n$:

$$x_i^t = \sum_{h<l\leq i\leq j} \left(w_{hlj}^t + u z_{hlj}^t\right) - \sum_{h\leq i<l\leq j} z_{hlj}^t + \sum_{h\leq i\leq j} u g_{hj}^t \tag{1}$$

- *For* $1 \leq i < j \leq m$:

$$f_{ij}^n = 0 = g_{ij}^n \tag{2}$$

- *For* $1 \leq i \leq j \leq m$ *and* $1 \leq t \leq n$:

$$f_{ij}^t + \sum_{i<l\leq j} w_{ilj}^t - \sum_{1\leq h<i} w_{hij}^t - p_{ij}^t - \sum_{j<h\leq m} r_{i(j+1)h}^t = 0 \tag{3}$$

$$g_{ij}^t + \sum_{i<l\leq j} z_{ilj}^t - \sum_{j<h\leq m} z_{i(j+1)h}^t - q_{ij}^t - \sum_{1\leq h<i} r_{hij}^t = 0 \tag{4}$$

- *For* $1 \leq i \leq j \leq m$ *and* $1 < t \leq n$:

$$p_{ij}^t + q_{ij}^t + \sum_{i<l\leq j} r_{ilj}^t - f_{ij}^{t-1} - g_{ij}^{t-1} - \sum_{j<h\leq m} w_{i(j+1)h}^{t-1} - \sum_{1\leq h<i} z_{hij}^{t-1} = 0 \tag{5}$$

- *For* $1 \leq i \leq j \leq m$:

$$p_{ij}^1 + q_{ij}^1 + \sum_{i<l\leq j} r_{ilj}^1 = \begin{cases} 0 & \text{if } (i,j) \neq (1,m) \\ 1 & \text{if } (i,j) = (1,m) \end{cases} \tag{6}$$

- $$f, w, g, z, p, q, r \geq 0. \tag{7}$$

Remark 5. *Note that (2)-(7) does not depend on* u*, which only appears in the projection (1).*

In order to prove Theorem 4, we first present a dynamic program to optimize a linear function over $L^u_{m,n}$, from which we derive an extended formulation for $\text{conv}(L^u_{m,n})$ following the approach of [12].

Given a linear objective function $c = \left[c_i^t\right] \in \mathbb{R}^{m\times n}$, consider the problem of computing $\mu := \min\left\{c \cdot x : \ x \in L^u_{m,n}\right\}$. Let c_{ij}^t denote the submatrix of c given by block (i,j,t) and let γ_{ij}^t be the optimal value over block (i,j,t), i.e., $\gamma_{ij}^t := \min\left\{c_{ij}^t \cdot x_{ij}^t : \ x_{ij}^t \in L^u_{j-i+1,n-t+1}\right\}$. Note that $\mu = \gamma_{1m}^1$. Also, let α_{ij}^t (resp. β_{ij}^t) be the optimal value over block (i,j,t) under the conditions that the first column of the solution begins with 0 (resp. ends with u) and does not have jumps.

Let w be the first column of an optimal solution in $L^u_{m,n}$ defining μ. By Proposition 3, we have the following cases. If w is constant, we can assume that all its entries are either equal to 0 or u, and the problem reduces to computing γ_{1m}^2 (cases (i) and (ii).) If w has splits but no jumps, then either $\mu = \alpha_{1m}^1$ or $\mu = \beta_{1m}^1$ (cases (iii) and (iv).) Finally, if w has a jump, then the problem reduces to

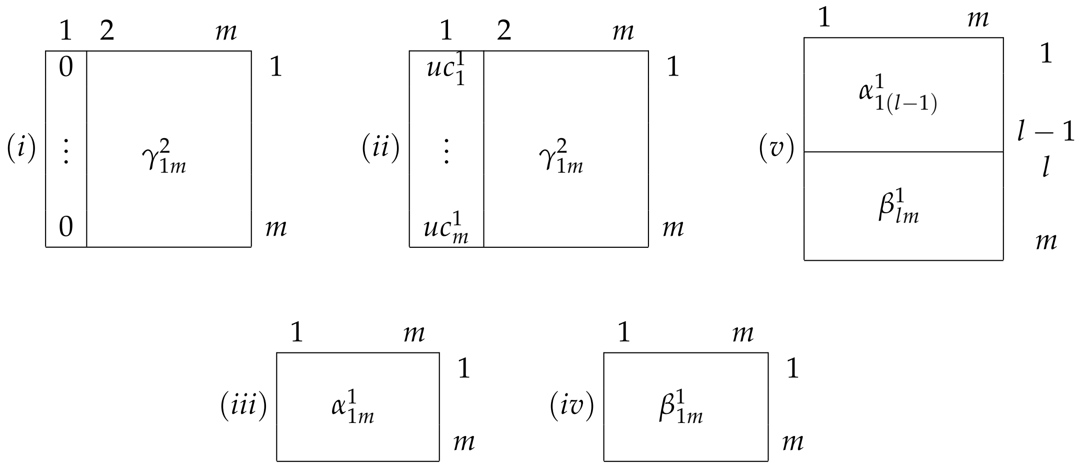

Figure 3: The five possibilities for $\mu = \gamma^1_{1m}$ given by a vertex $x = x^1_{1m}$ of $\mathrm{conv}(L^u_{m,n})$.

computing the optimal position of the jump, i.e., minimizing $\alpha^1_{1(l-1)} + \beta^1_{lm}$ over $1 < l \leq m$ (case (v).) The costs of these cases are shown in Figure 3.

Assuming $u \geq m-1$, consider the following system on α, β, γ:

- For $1 \leq i \leq m$:

$$\alpha^n_{ii} = 0 \tag{8a}$$

$$\beta^n_{ii} = uc^n_i \tag{8b}$$

- For $1 \leq i < j \leq m$:

$$\alpha^n_{ij} = \min_{i<l\leq j} \left\{ 0 + \alpha^n_{lj} + \sum_{h=l}^{j} c^n_h \right\} \tag{9a}$$

$$\beta^n_{ij} = \min_{i<l\leq j} \left\{ \beta^n_{i(l-1)} - \sum_{h=i}^{l-1} c^n_h + u \sum_{h=l}^{j} c^n_h \right\} \tag{9b}$$

- For $1 \leq i \leq j \leq m$ and $1 \leq t < n$:

$$\alpha^t_{ij} = \min \left\{ 0 + \gamma^{t+1}_{ij},\ \min_{i<l\leq j} \left\{ 0 + \gamma^{t+1}_{i(l-1)} + \alpha^t_{lj} + \sum_{h=l}^{j} c^t_h \right\} \right\} \tag{10a}$$

$$\beta^t_{ij} = \min \left\{ u \sum_{h=i}^{j} c^t_h + \gamma^{t+1}_{ij},\ \min_{i<l\leq j} \left\{ \beta^t_{i(l-1)} - \sum_{h=i}^{l-1} c^t_h + u \sum_{h=l}^{j} c^t_h + \gamma^{t+1}_{lj} \right\} \right\} \tag{10b}$$

- For $1 \leq i \leq j \leq m$ and $1 \leq t \leq n$:

$$\gamma^t_{ij} = \min \left\{ \alpha^t_{ij},\ \beta^t_{ij},\ \min_{i<l\leq j} \left\{ \alpha^t_{i(l-1)} + \beta^t_{lj} \right\} \right\}. \tag{11}$$

Equation (9a) picks an index l to place a 1 (increase by 1), shifting the cost over (l, j, n) by $\sum_{h=l}^{j} c_h^n$ with respect to α_{lj}^n. Similarly, equation (9b) picks an index $l-1$ to place a $u-1$ (decrease by 1), shifting the cost over $(i, l-1, n)$ by $-\sum_{h=i}^{l-1} c_h^n$ with respect to $\beta_{i(l-1)}^n$. The condition $u \geq m-1$ ensures that values do not overlap while applying these two operations. The same idea applies to the inner minimization in (10a) and (10b), which correspond to cases (iii) and (iv) of Proposition 3 and are depicted in Figure 4 for α_{1m}^1 and β_{1m}^1.

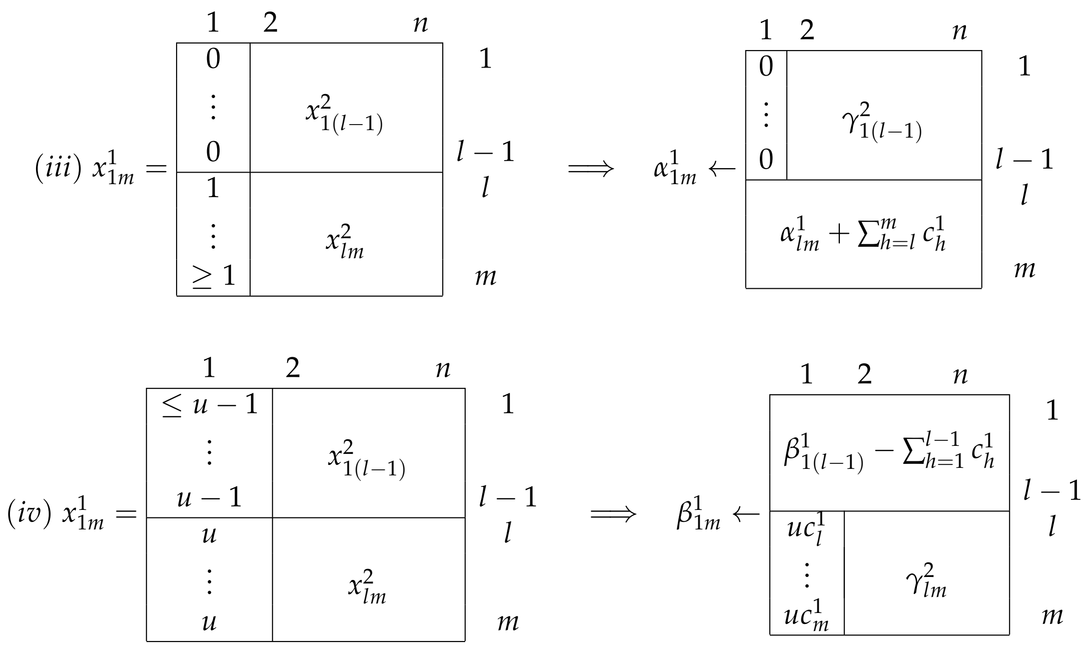

Figure 4: Computing α_{1m}^1 and β_{1m}^1.

Proposition 6. *The system (9a)-(11) is consistent with the definitions of α_{ij}^t, β_{ij}^t and γ_{ij}^t for $1 \leq i \leq j \leq m$ and $1 \leq t \leq n$.*

Proof. Let $\mathbf{0}$ and $\mathbf{1}$ denote vectors of 0's and 1's of appropriate size, respectively.

- For $1 \leq i \leq m$, x_{ii}^n consists of a single variable. If $\hat{x}_{ii}^n$ is an optimal solution defining α_{ii}^n, then the only option is $\hat{x}_{ii}^n = 0$, while for β_{ii}^n, the only option is $\hat{x}_{ii}^n = u$. Then (8a) and (8b) follow.
- For $1 \leq i < j \leq m$, x_{ij}^n consists of a single column of height at least 2.

  Let $\hat{x}_{ij}^n$ be an optimal solution defining α_{ij}^n. Since $\hat{x}_{ij}^n$ must begin with 0 but cannot be constant, there exists $i < l \leq j$ such that $\hat{x}_{i(l-1)}^n = \mathbf{0}$ and $\hat{x}_{ll}^n = 1$. Since $\hat{x}_{lj}^n - \mathbf{1}$ begins with 0 and does not have jumps, it must be an optimal solution defining α_{lj}^n, for otherwise $\hat{x}_{ij}^n$ would not be optimal for α_{ij}^n. This implies $c_{lj}^n \cdot \hat{x}_{lj}^n = c_{lj}^n \cdot (\hat{x}_{lj}^n - \mathbf{1} + \mathbf{1}) = \alpha_{lj}^n + \sum_{h=l}^{j} c_h^n$. Since $c_{i(l-1)}^n \cdot \hat{x}_{i(l-1)}^n = 0$, (9a) follows.

  Similarly, let $\hat{x}_{ij}^n$ be an optimal solution defining β_{ij}^n. Since $\hat{x}_{ij}^n$ must end with u but cannot be constant, there exists $i < l \leq j$ such that $\hat{x}_{lj}^n = u\mathbf{1}$ and $\hat{x}_{(l-1)(l-1)}^n = u-1$. Since $\hat{x}_{i(l-1)}^n + \mathbf{1}$ ends with u and does not have jumps, it must be an optimal solution defining $\beta_{i(l-1)}^n$, for otherwise

$\hat{x}^n_{ij}$ would not be optimal for β^n_{ij}. This implies $c^n_{i(l-1)} \cdot \hat{x}^n_{i(l-1)} = c^n_{i(l-1)} \cdot (\hat{x}^n_{i(l-1)} + \mathbf{1} - \mathbf{1}) = \beta^n_{i(l-1)} - \sum_{h=i}^{l-1} c^n_h$. Since $c^n_{lj} \cdot \hat{x}^n_{lj} = u \sum_{h=l}^{j} c^n_h$, (9b) follows.

- For $1 \le i \le j \le m$ and $1 \le t < n$, x^t_{ij} is a matrix with two columns at least. We reason as before.

  Let $\hat{x}^t_{ij}$ be an optimal solution defining α^t_{ij}. Then, either its first column is equal to $\mathbf{0}$, or it has a first split at some $i < l \le j$. In the former case, $\hat{x}^{t+1}_{ij}$ must be an optimal solution defining γ^{t+1}_{ij}, and thus $\alpha^t_{ij} = 0 + \gamma^{t+1}_{ij}$. In the latter, $\hat{x}^t_{i(l-1)}$ and $\hat{x}^t_{lj}$ define optimal solutions to two subproblems. The first one, since the first column of $\hat{x}^t_{i(l-1)}$ is equal to $\mathbf{0}$, has optimal value $0 + \gamma^{t+1}_{i(l-1)}$. The second one, since the first column of $\hat{x}^t_{lj}$ begins with 1 and does not have jumps, has optimal value $\alpha^t_{lj} + \sum_{h=l}^{j} c^t_h$. Therefore, either $\alpha^t_{ij} = 0 + \gamma^{t+1}_{ij}$, or $\alpha^t_{ij} = 0 + \gamma^{t+1}_{i(l-1)} + \alpha^t_{lj} + \sum_{h=l}^{j} c^t_h$ for some $i < l \le j$. This gives us (10a).

  Similarly, let $\hat{x}^t_{ij}$ be an optimal solution defining β^t_{ij}. Then, either its first column is equal to $u\mathbf{1}$, or it has a last split at some $i < l \le j$. In the former case, $\hat{x}^{t+1}_{ij}$ must be an optimal solution defining γ^{t+1}_{ij}, and thus $\beta^t_{ij} = u \sum_{h=i}^{j} c^t_h + \gamma^{t+1}_{ij}$. In the latter, $\hat{x}^t_{i(l-1)}$ and $\hat{x}^t_{lj}$ define optimal solutions to two subproblems. The first one, since the first column of $\hat{x}^t_{i(l-1)}$ ends with $u - 1$ and does not have jumps, has optimal value $\beta^t_{i(l-1)} - \sum_{h=i}^{l-1} c^t_h$. The second one, since the first column of $\hat{x}^t_{lj}$ is equal to $u\mathbf{1}$, has optimal value $u \sum_{h=l}^{j} c^t_h + \gamma^{t+1}_{lj}$. Therefore, either $\beta^t_{ij} = u \sum_{h=i}^{j} c^t_h + \gamma^{t+1}_{ij}$, or $\beta^t_{ij} = \beta^t_{i(l-1)} - \sum_{h=i}^{l-1} c^t_h + u \sum_{h=l}^{j} c^t_h + \gamma^{t+1}_{lj}$ for some $i < l \le j$. This gives us (10b).

- For $1 \le i \le j \le m$ and $1 \le t \le n$, let $\hat{x}^t_{ij}$ be an optimal solution defining γ^t_{ij}. If its first column does not have jumps, then $\gamma^t_{ij} = \alpha^t_{ij}$ or $\gamma^t_{ij} = \beta^t_{ij}$. Otherwise, $\gamma^t_{ij} = \alpha^t_{i(l-1)} + \beta^t_{lj}$ for some $i < l \le j$. Note that if $\alpha^t_{i(l-1)}$ is given by $\hat{x}^t_{i(l-1)}$, then $\hat{x}^t_{l-1} \le l - 2$, and if β^t_{lj} is given by $\hat{x}^t_{lj}$, then $\hat{x}^t_l \ge u - (m - l) \ge l - 1$. Finally, we obtain (11).

□

In linear programming form, from Proposition 6, we obtain that μ is equal to the maximum value of γ^1_{1m} over (α, β, γ) such that

- For $1 \le i \le m$:

$$\alpha^n_{ii} \le 0 \quad (f^n_{ii}) \tag{12a}$$

$$\beta^n_{ii} \le u c^n_i \quad (g^n_{ii}) \tag{12b}$$

- For $1 \le i < j \le m$:

$$\alpha^n_{ij} \le \alpha^n_{lj} + \sum_{h=l}^{j} c^n_h \quad i < l \le j \quad (w^n_{ilj}) \tag{13a}$$

$$\beta^n_{ij} \le \beta^n_{i(l-1)} - \sum_{h=i}^{l-1} c^n_h + u \sum_{h=l}^{j} c^n_h \quad i < l \le j \quad (z^n_{ilj}) \tag{13b}$$

- For $1 \leq i \leq j \leq m$ and $1 \leq t < n$:

$$\alpha_{ij}^t \leq \begin{cases} \gamma_{ij}^{t+1} & & (f_{ij}^t) \\ \gamma_{i(l-1)}^{t+1} + \alpha_{lj}^t + \sum_{h=l}^{j} c_h^t & i < l \leq j & (w_{ilj}^t) \end{cases} \tag{14a}$$

$$\beta_{ij}^t \leq \begin{cases} \gamma_{ij}^{t+1} + u \sum_{h=i}^{j} c_h^t & & (g_{ij}^t) \\ \beta_{i(l-1)}^t - \sum_{h=i}^{l-1} c_h^t + u \sum_{h=l}^{j} c_h^t + \gamma_{lj}^{t+1} & i < l \leq j & (z_{ilj}^t) \end{cases} \tag{14b}$$

- For $1 \leq i \leq j \leq m$ and $1 \leq t \leq n$:

$$\gamma_{ij}^t \leq \begin{cases} \alpha_{ij}^t & & (p_{ij}^t) \\ \beta_{ij}^t & & (q_{ij}^t) \\ \alpha_{i(l-1)}^t + \beta_{lj}^t & i < l \leq j & (r_{ilj}^t) \end{cases} \tag{15}$$

In view of (12a)-(15), we can define a hypergraph representing our dynamic program. The set S of states is identified with the set of variables in the vector (α, β, γ). For each inequality in (12a)-(15), we include in H an hyperarc of the form (Δ, s), with s the variable on the left-hand side and Δ the set of variables on the right-hand side. Now we are ready to prove Theorem 4.

of Theorem 4. Defining $\pi := (\alpha, \beta, \gamma)$ and $\pi_0 := \gamma_{1m}^1$, the problem $\max\{\gamma_{1m}^1 : \ (12a) - (15)\}$ can be written as $\max\{\pi_0 : \ A^\top \pi \leq d\}$ for appropriate constraint matrix A and right-hand side vector d. If we consider this linear program as the dual of another linear program, then the corresponding primal problem is $\min\{d^\top y : \ Ay = e_0, \ y \geq 0\}$, with e_0 the canonical vector with a 1 at the position given by π_0. More precisely, the primal problem is to minimize

$$\sum_{\substack{1 \leq t \leq n \\ 1 \leq i \leq j \leq m}} \left[g_{ij}^t u \sum_{h=i}^{j} c_h^t + \sum_{i < l \leq j} \left(w_{ilj}^t \sum_{h=l}^{j} c_h^t + z_{ilj}^t \left(- \sum_{h=i}^{l-1} c_h^t + u \sum_{h=l}^{j} c_h^t \right) \right) \right] \tag{16}$$

over $y := (f, w, g, z, p, q, r)$ satisfying (2)-(7), where for $1 \leq i < j \leq m$, we have introduced additional variables f_{ij}^n and g_{ij}^n, both fixed to 0, to simplify the formulation. Grouping terms in (16) with respect to c_i^t, we obtain that x_i^t is given by (1), which maps y onto x.

Finally, to ensure that (2)-(7) has binary vertices only, and thus the polyhedral characterization of the dynamic program is correct, we need a reference set as in [12]. In our case, it suffices to take I as the complete block of indices, that is, $I = (1, m, 1)$, and tag states $\alpha_{ij}^t, \beta_{ij}^t, \gamma_{ij}^t$ with block (i, j, t). □

4 The case $1 < u < m - 1$

The main result of this section is the following.

Theorem 7. *The system (17)-(23) below is an extended formulation for* $\operatorname{conv}(L_{m,n}^u)$ *with* $\mathcal{O}(um^3n)$ *variables and constraints:*

- *For $1 \le i \le m$ and $1 \le t \le n$:*

$$x_i^t = \sum_{\substack{h<l\le i\le j\\ 0<v\le u}} w_{hlj}^{vt} + u \sum_{\substack{h<l\le i\le j\\ 0\le v<u}} z_{hlj}^{vt} - \sum_{\substack{h\le i<l\le j\\ 0\le v<u}} z_{hlj}^{vt} + \sum_{\substack{h\le i\le j\\ 0\le v\le u}} u g_{hj}^{vt} \tag{17}$$

- *For $1 \le i < j \le m$ and $0 \le v \le u$:*

$$f_{ij}^{vn} = 0 = g_{ij}^{vn} \tag{18}$$

- *For $1 \le i \le j \le m$, $1 \le t \le n$, and $0 \le v \le u$:*

$$f_{ij}^{vt} = \begin{cases} \sum_{1\le h<i} w_{hij}^{1t} + \sum_{j<h\le m} r_{i(j+1)h}^{1t} & \text{if } v = 0 \\ \sum_{1\le h<i} w_{hij}^{(v+1)t} + \sum_{j<h\le m} r_{i(j+1)h}^{(v+1)t} - \sum_{i<l\le j} w_{ilj}^{vt} & \text{if } 0 < v < u \\ p_{ij}^t - \sum_{i<l\le j} w_{ilj}^{ut} & \text{if } v = u \end{cases} \tag{19}$$

$$g_{ij}^{vt} = \begin{cases} q_{ij}^t - \sum_{i<l\le j} z_{ilj}^{0t} & \text{if } v = 0 \\ \sum_{j<h\le m} z_{i(j+1)h}^{(v-1)t} + \sum_{1\le h<i} r_{hij}^{(v-1)t} - \sum_{i<l\le j} z_{ilj}^{vt} & \text{if } 0 < v < u \\ \sum_{j<h\le m} z_{i(j+1)h}^{(u-1)t} + \sum_{1\le h<i} r_{hij}^{(u-1)t} & \text{if } v = u \end{cases} \tag{20}$$

- *For $1 \le i \le j \le m$ and $1 < t \le n$:*

$$p_{ij}^t + q_{ij}^t + \sum_{\substack{i<l\le j\\ 0<v<u}} r_{ilj}^{vt} = \sum_{0\le v\le u} \left(f_{ij}^{v(t-1)} + g_{ij}^{v(t-1)} \right) + \sum_{\substack{j<h\le m\\ 0<v\le u}} w_{i(j+1)h}^{v(t-1)} + \sum_{\substack{1\le h<i\\ 0\le v<u}} z_{hij}^{v(t-1)} \tag{21}$$

- *For $1 \le i \le j \le m$:*

$$p_{ij}^1 + q_{ij}^1 + \sum_{\substack{i<l\le j\\ 0<v<u}} r_{ilj}^{v1} = \begin{cases} 0 & \text{if } (i,j) \ne (1,m) \\ 1 & \text{if } (i,j) = (1,m) \end{cases} \tag{22}$$

- $$f, w, g, z, p, q, r \ge 0. \tag{23}$$

Remark 8. *Note that the number of variables and constraints in (18)-(23) grows linearly with u.*

The proof of Theorem 7 is along the lines of that of Theorem 4. However, if $1 < u < m-1$, then the dynamic program has to be modified to avoid overlapping. We introduce an index v to track bounds. Let α_{ij}^{vt} (resp. β_{ij}^{vt}) be the optimal value over block (i,j,t) under the conditions that the first column of the solution begins with 0 (resp. ends with u), its entries are less than or equal to v (resp. greater than or equal to v), and does not have jumps. As before, let γ_{ij}^t be the optimal value over block (i,j,t). Consider the following system on α, β, γ:

- For $1 \le i \le m$:

$$\alpha_{ii}^{vn} = 0 \quad 0 \le v \le u \tag{24a}$$

$$\beta_{ii}^{vn} = uc_i^n \quad 0 \le v \le u \tag{24b}$$

- For $1 \leq i < j \leq m$:

$$\alpha_{ij}^{vn} = \min_{i<l\leq j} \left\{ 0 + \alpha_{lj}^{(v-1)n} + \sum_{h=l}^{j} c_h^n \right\} \quad 0 < v \leq u \tag{25a}$$

$$\alpha_{ij}^{0n} = +\infty \tag{25b}$$

$$\beta_{ij}^{vn} = \min_{i<l\leq j} \left\{ \beta_{i(l-1)}^{(v+1)n} - \sum_{h=i}^{l-1} c_h^n + u \sum_{h=l}^{j} c_h^n \right\} \quad 0 \leq v < u \tag{25c}$$

$$\beta_{ij}^{un} = +\infty \tag{25d}$$

- For $1 \leq i \leq j \leq m$ and $1 \leq t < n$:

$$\alpha_{ij}^{vt} = \min \left\{ 0 + \gamma_{ij}^{t+1},\ \min_{i<l\leq j} \left\{ 0 + \gamma_{i(l-1)}^{t+1} + \alpha_{lj}^{(v-1)t} + \sum_{h=l}^{j} c_h^t \right\} \right\} \quad 0 < v \leq u \tag{26a}$$

$$\alpha_{ij}^{0t} = 0 + \gamma_{ij}^{t+1} \tag{26b}$$

$$\beta_{ij}^{vt} = \min \left\{ u \sum_{h=i}^{j} c_h^t + \gamma_{ij}^{t+1},\ \min_{i<l\leq j} \left\{ \beta_{i(l-1)}^{(v+1)t} - \sum_{h=i}^{l-1} c_h^t + u \sum_{h=l}^{j} c_h^t + \gamma_{lj}^{t+1} \right\} \right\} \quad 0 \leq v < u \tag{26c}$$

$$\beta_{ij}^{ut} = u \sum_{h=i}^{j} c_h^t + \gamma_{ij}^{t+1} \tag{26d}$$

- For $1 \leq i \leq j \leq m$ and $1 \leq t \leq n$:

$$\gamma_{ij}^{t} = \min \left\{ \alpha_{ij}^{ut},\ \beta_{ij}^{0t},\ \min_{i<l\leq j,\ 0<v<u} \left\{ \alpha_{i(l-1)}^{(v-1)t} + \beta_{lj}^{(v+1)t} \right\} \right\}. \tag{27}$$

Proposition 9. *The system (24a)-(27) is consistent with the definitions of α_{ij}^{vt}, β_{ij}^{vt} and γ_{ij}^{t} for $1 \leq i \leq j \leq m$, $1 \leq t \leq n$, and $0 \leq v \leq u$.*

From Proposition 9, whose proof is omitted for brevity, we can proceed in a similar way as we did for the case $u \geq m-1$ by considering a dual linear program and its primal problem.

- For $1 \leq i \leq m$:

$$\alpha_{ii}^{vn} \leq 0 \quad 0 \leq v \leq u \quad (f_{ii}^{vn})$$
$$\beta_{ii}^{vn} \leq u c_i^n \quad 0 \leq v \leq u \quad (g_{ii}^{vn})$$

- For $1 \leq i < j \leq m$:

$$\alpha_{ij}^{vn} \leq \alpha_{lj}^{(v-1)n} + \sum_{h=l}^{j} c_h^n \quad i < l \leq j,\ 0 < v \leq u \quad (w_{ilj}^{vn})$$

$$\beta_{ij}^{vn} \leq \beta_{i(l-1)}^{(v+1)n} - \sum_{h=i}^{l-1} c_h^n + u \sum_{h=l}^{j} c_h^n \quad i < l \leq j,\ 0 \leq v < u \quad (z_{ilj}^{vn})$$

- For $1 \leq i \leq j \leq m$ and $1 \leq t < n$:

$$\alpha_{ij}^{vt} \leq \begin{cases} \gamma_{ij}^{t+1} & & (f_{ij}^{vt}) \\ \gamma_{i(l-1)}^{t+1} + \alpha_{lj}^{(v-1)t} + \sum_{h=l}^{j} c_h^t & i < l \leq j & (w_{ilj}^{vt}) \end{cases} \quad 0 < v \leq u$$

$$\alpha_{ij}^{0t} \leq \gamma_{ij}^{t+1} \quad (f_{ij}^{0t})$$

$$\beta_{ij}^{vt} \leq \begin{cases} \gamma_{ij}^{t+1} + u \sum_{h=i}^{j} c_h^t & & (g_{ij}^{vt}) \\ \beta_{i(l-1)}^{(v+1)t} - \sum_{h=i}^{l-1} c_h^t + u \sum_{h=l}^{j} c_h^t + \gamma_{lj}^{t+1} & i < l \leq j & (z_{ilj}^{vt}) \end{cases} \quad 0 \leq v < u$$

$$\beta_{ij}^{ut} \leq \gamma_{ij}^{t+1} + u \sum_{h=i}^{j} c_h^t \quad (g_{ij}^{ut})$$

- For $1 \leq i \leq j \leq m$ and $1 \leq t \leq n$:

$$\gamma_{ij}^{t} \leq \begin{cases} \alpha_{ij}^{ut} & & (p_{ij}^{t}) \\ \beta_{ij}^{0t} & & (q_{ij}^{t}) \\ \alpha_{i(l-1)}^{(v-1)t} + \beta_{lj}^{(v+1)t} & i < l \leq j,\ 0 < v < u & (r_{ilj}^{vt}) \end{cases}$$

Then Theorem 7 follows from linear programming duality and the approach of [12], where for $1 \leq i < j \leq m$ and $0 \leq v \leq u$, we have introduced additional variables f_{ij}^{vn} and g_{ij}^{vn}, both fixed to 0, to simplify the formulation.

References

[1] G. Angulo, S. Ahmed, S.S. Dey, and V. Kaibel, *Forbidden vertices*, Math. of Oper. Res. **40** (2014), no. 2, 350–360.

[2] M. Barbato, R. Grappe, M. Lacroix, and C. Pira, *Lexicographical polytopes*, Discret. Appl. Math. **240** (2018), 3–7.

[3] M. Di Summa, *A short convex-hull proof for the all-different system with the inclusion property*, Oper. Res. Lett. **43** (2015), no. 1, 69–73.

[4] A. Gupte, *Convex hulls of superincreasing knapsacks and lexicographic orderings*, Discret. Appl. Math. **201** (2016), 150–163.

[5] C. Hojny and M.E. Pfetsch, *Polytopes associated with symmetry handling*, Math. Program. (2018).

[6] V. Kaibel and A. Loos, *Branched polyhedral systems*, Integer Programming and Combinatorial Optimization, Springer, 2010, pp. 177–190.

[7] V. Kaibel and M.E. Pfetsch, *Packing and partitioning orbitopes*, Math. Program. **114** (2008), no. 1, 1–36.

[8] J. Lee, *All-different polytopes*, J. of Comb. Optim. **6** (2002), no. 3, 335–352.

[9] J. Lee and F. Margot, *On a binary-encoded ILP coloring formulation*, INFORMS J. on Comput. **19** (2007), no. 3, 406–415.

[10] A. Loos, *Describing Orbitopes by Linear Inequalities and Projection Based Tools*, Ph.D. thesis, 2011.

[11] D. Magos, I. Mourtos, and G. Appa, *A polyhedral approach to the alldifferent system*, Math. Program. **132** (2012), no. 1-2, 209–260.

[12] R.K. Martin, R.L. Rardin, and B.A. Campbell, *Polyhedral characterization of discrete dynamic programming*, Oper. Res. **38** (1990), no. 1, 127–138.

[13] A. Schrijver, *Theory of linear and integer programming*, Wiley, 1998.

[14] H.P. Williams and H. Yan, *Representations of the all_different predicate of constraint satisfaction in integer programming*, INFORMS J. on Comput. **13** (2001), no. 2, 96–103.