5-2019

Finding the Symmetry Group of an LP with Equality Constraints and Its Application to Classifying Orthogonal Arrays

Andrew J. Geyer
Air Force Institute of Technology

Dursun A. Bulutoglu
Air Force Institute of Technology

Kenneth J. Ryan
West Virginia University

Recommended Citation

Geyer, A. J., Bulutoglu, D. A., & Ryan, K. J. (2019). Finding the symmetry group of an LP with equality constraints and its application to classifying orthogonal arrays. Discrete Optimization, 32, 93–119. https://doi.org/10.1016/j.disopt.2019.01.001

Finding the symmetry group of an LP with equality constraints and its application to classifying orthogonal arrays

Andrew J. Geyer[a], Dursun A. Bulutoglu[a,*], Kenneth J. Ryan[b]

[a] *Department of Mathematics and Statistics, Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio 45433, USA*
[b] *Department of Statistics, West Virginia University, Morgantown, West Virginia 26506, USA*

Abstract

For a given linear program (LP) a permutation of its variables that sends feasible points to feasible points and preserves the objective function value of each of its feasible points is a symmetry of the LP. The set of all symmetries of an LP, denoted by G^{LP}, is the symmetry group of the LP. Margot [F. Margot, 50 Years of Integer Programming 1958-2008 (2010), 647-686] described a method for computing a subgroup of the symmetry group G^{LP} of an LP. This method computes G^{LP} when the LP has only non-redundant inequalities and its feasible set satisfies no equality constraints. However, when the feasible set of the LP satisfies equality constraints this method finds only a subgroup of G^{LP} and can miss symmetries. We develop a method for finding the symmetry group of a feasible LP whose feasible set satisfies equality constraints. We apply this method to find and exploit the previously unexploited symmetries of an orthogonal array defining integer linear program (ILP) within the branch-and-bound (B&B) with isomorphism pruning algorithm [F. Margot, Symmetric ILP: Coloring and small integers, Discrete Optimization 4 (1) (2007), 40-62]. Our method reduced the running time for finding all OD-equivalence classes of $\text{OA}(160, 8, 2, 4)$ and $\text{OA}(176, 8, 2, 4)$ by factors of $1/(2.16)$ and $1/(1.36)$ compared to the fastest known method [D. A. Bulutoglu and K. J. Ryan, Integer programming for classifying orthogonal arrays, Australasian Journal of Combinatorics 70 (3) (2018), 362-385]. These were the two bottleneck cases that could not have been solved until the B&B with isomorphism pruning algorithm was applied. Another key finding of this paper is that converting inequalities to equalities by introducing slack variables and exploiting the symmetry group of the resulting ILP's LP relaxation within the B&B with isomorphism pruning algorithm can reduce the computation time by several orders of magnitude when enumerating a set of all non-isomorphic solutions of an ILP.

Keywords: Vertex colored, edge colored graph; Formulation symmetry group; LP relaxation symmetry group; OD-equivalence; Orthogonal projection matrix
2000 MSC: 90C05 90C10 68R10

1. Introduction

A branch-and-bound (B&B) algorithm can be used to find an optimum solution or enumerate all optimum solutions to an integer linear program (ILP) of the form

$$\begin{aligned} \min \quad & \mathbf{c}^T\mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} = \mathbf{b}, \quad \mathbf{x} \in \mathbb{Z}^n, \\ & \mathbf{B}\mathbf{x} \le \mathbf{d}. \end{aligned} \tag{1}$$

*Corresponding author
Email addresses: andrew.geyer@afit.edu (Andrew J. Geyer), dursun.bulutoglu@gmail.com (Dursun A. Bulutoglu), kjryan@mail.wvu.edu (Kenneth J. Ryan)

Preprint submitted to Discrete Optimization *December 31, 2018*

Let $\mathbf{x}$ be called a *partial solution* of ILP (1) if each element of a strict subset of entries of $\mathbf{x}$ has been assigned integer values and the remaining entries are not fixed. One way a B&B algorithm that branches on the integer values of the variables of $\mathbf{x}$ prunes a partial solution, i.e., a node of its backtrack search tree, is by infeasibility (pruning by infeasibility). A partial solution is pruned by infeasibility if the linear programming (LP) relaxation of the *subproblem* created from ILP (1) by assigning the fixed integer values in the partial solution $\mathbf{x}$ to their corresponding variables is proven to be infeasible by solving the LP relaxation. Another way of pruning is by comparing the optimum LP relaxation value of a subproblem to that of the best known solution. If this objective function value is worse or the same, then the node corresponding to the partial solution that created the subproblem is pruned (pruning by bound). A third way of pruning is when a subproblem is solved, i.e., an integral solution with the objective function value matching the optimum LP relaxation value of the subproblem is found (pruning by optimality). The LP relaxation of the problem at the root node is solved using the primal simplex algorithm [28]. The LP relaxations of the subproblems created at the non-root nodes are solved using the dual simplex algorithm taking advantage of warm starts [2]. Every time B&B finds a solution with a better objective function value than that of the incumbent best solution, the best solution is updated with the new solution. If the goal is to find an optimum solution, then B&B can be stopped as soon as it finds a solution whose objective function value is equal to the best known lower bound for the optimum value of ILP (1) because this solution must be optimum. For more details, see Chapter 7 of [36].

All optimum solutions of ILP (1) can be enumerated by using a depth-first search B&B that branches on the integer values of the variables if the optimum value of ILP (1) is known in advance and say is equal to z^*. This is done by pruning a partial solution corresponding to a subproblem if and only if z^* is strictly smaller than the optimum LP relaxation value of the subproblem or the subproblem is infeasible. When enumerating all optimum solutions no partial solution is pruned by optimality. This version of the B&B algorithm was used in [10] to classify orthogonal arrays $\mathrm{OA}(N, k, s, t)$ up to isomorphism and a weaker form of isomorphism for many N, k, s, t combinations. It was also used in [5, 6, 17] to classify all non-isomorphic $\mathrm{OA}(N, k, s, t)$, covering arrays with the minimum number of rows $\mathrm{CA}^*_\lambda(k, s, t)$, packing arrays with the maximum number of rows $\mathrm{PA}^*_\lambda(k, s, t)$, 4-(10, 5, 1)-covering designs with the minimum number of sets (blocks), and all $\mathrm{OA}(N, k, 2, t)$ up to OD-equivalence for many N, k, s, t combinations. However, the bottleneck classifications of all $\mathrm{OA}(N, k, 2, t)$ up to OD-equivalence in [6] required using `nauty` [25, 26] to remove OD-equivalent $\mathrm{OA}(N, k, 2, t)$. In this paper, we also use this version of a B&B algorithm to directly classify all bottleneck $\mathrm{OA}(N, k, 2, t)$ in [6] up to OD-equivalence without resorting to `nauty` [25, 26] for removing OD-equivalent $\mathrm{OA}(N, k, 2, t)$. (The definitions of $\mathrm{OA}(N, k, s, t)$, isomorphism of $\mathrm{OA}(N, k, s, t)$, and OD-equivalence of $\mathrm{OA}(N, k, 2, t)$ are deferred until Section 4.) Throughout this paper, when we refer to a B&B algorithm we mean a depth-first search B&B that branches on the integer values of the variables targeted to find all optimum solutions of an ILP.

The group of all permutations of the variables of ILP (1) that map feasible points onto feasible points and preserve the objective function value of each feasible point is called the *symmetry group* of ILP (1) [20, 29]. For a subgroup G of the symmetry group of ILP (1), two (partial) solutions $\mathbf{x}_1$ and $\mathbf{x}_2$ of ILP (1) are called *isomorphic under the action of* G if $g(\mathbf{x}_1) = \mathbf{x}_2$ for some $g \in G$, where

$$g((x_1, \ldots, x_n)^T) = (x_{g^{-1}(1)}, \ldots, x_{g^{-1}(n)})^T.$$

Similarly, two subproblems of ILP (1) are called *isomorphic subproblems* if they are created from isomorphic partial solutions. Clearly, the LP relaxation of isomorphic subproblems have the same optimum objective function value and feasibility status. Hence, when the symmetry group of ILP (1) is large, a B&B algorithm wastes time by solving the LP relaxations of a large number of isomorphic subproblems created from the same number of isomorphic partial solutions. To address this issue, Margot [16, 17, 18, 19] developed the B&B with isomorphism pruning algorithm that finds a set of all non-isomorphic optimal solutions to ILP (1) by solving the LP relaxation of only the unique subproblem created from the unique lexicographically minimum partial solution under the action of G, where the lexicographical ordering of partial solutions is defined as follows.

Definition 1. Let $\mathbf{x}$ and $\mathbf{x}'$ be two partial solutions of ILP (1) in a B&B search tree. Let $i_1, \ldots, i_{r_1}$ and $i'_1, \ldots, i'_{r_2}$ be the indices of the variables in $\mathbf{x}$ and $\mathbf{x}'$ that are fixed by branching decisions, and $\gamma = \min\{r_1, r_2\}$. We say that $\mathbf{x}$ is *lexicographically smaller* than $\mathbf{x}'$ if one of the following two conditions is satisfied.

1. The first non-zero entry in $(i_1 - i'_1, \ldots, i_\gamma - i'_\gamma)$ is negative.
2. $(i_1, \ldots, i_\gamma) = (i'_1, \ldots, i'_\gamma)$ and the first non-zero entry in $(x_{i_1} - x'_{i'_1}, \ldots, x_{i_\gamma} - x'_{i'_\gamma})$ is positive.

When a B&B algorithm always selects the minimum index non-fixed variable for branching (called *minimum index branching*), then removing a partial solution (node) of the B&B search tree if it is not lexicographically minimum under the action of G results in a B&B tree whose all feasible leaves are a set of all non-isomorphic optimal solutions [19]. This is true because when minimum index branching is implemented, each lexicographically minimum node under the action of G has a unique lexicographically minimum parent node under the same action. Margot [19] developed an algorithm based on group theory to decide whether a partial solution is lexicographically minimum under the action of G. This algorithm is used within B&B with isomorphism pruning to prune isomorphic partial solutions.

Even when relatively small size groups are used, the search tree of B&B with isomorphism pruning is much smaller than that of B&B only, causing huge reductions in computation times [19]. However, to correctly find or classify solutions that are optimal or prove infeasibility, it is necessary that the symmetry group of ILP (1) or one of its subgroups is used. Whether two subproblems are deemed isomorphic depends on the subgroup used within B&B with isomorphism pruning. Subproblems that are inherently isomorphic may be deemed not to be isomorphic if a smaller subgroup is used. Consequently, using a larger subgroup results in a B&B tree with a smaller number of nodes where LP relaxations must be solved. Hence, it is desirable to find the symmetry group of a given ILP. If this is not possible, finding larger subgroups is more desirable (finding the symmetry group of an ILP is an NP-hard problem [20]). One subgroup of the symmetry group of an ILP is the *formulation symmetry group.* Finding this group is as hard as the graph isomorphism problem, which is not known to be solvable in polynomial time. The formulation symmetry group of ILP (1) given in [1] is defined to be

$$G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{c}) = \left\{ \pi \mid \pi(\mathbf{c}) = \mathbf{c},\ \exists \sigma \text{ with } \mathbf{A}(\pi, \sigma) = \mathbf{A},\ \mathbf{B}(\pi, \sigma) = \mathbf{B},\ \sigma \begin{bmatrix} \mathbf{b} \\ \mathbf{d} \end{bmatrix} = \begin{bmatrix} \mathbf{b} \\ \mathbf{d} \end{bmatrix} \right\}, \quad (2)$$

where $\mathbf{A}(\pi, \sigma)$ is the matrix obtained by permuting the columns of $\mathbf{A}$ with π followed by a permutation of its rows with σ. For a more general definition that covers mixed integer non-linear programs (MINLPs), see [13]. Define the formulation symmetry group of a generic LP (1) to be $G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{c})$, where LP (1) is obtained by dropping the $\mathbf{x} \in \mathbb{Z}^n$ constraint in ILP (1). Throughout the paper, whenever ILP (1) is referred to as LP (1) it is understood that LP (1) is obtained by dropping the integrality constraints in ILP (1). If ILP (1) or LP (1) has no equality constraints, then define its formulation symmetry group to be

$$G(\mathbf{B}, \mathbf{d}, \mathbf{c}) = \{\pi \mid \pi(\mathbf{c}) = \mathbf{c},\ \exists \sigma \text{ with } \mathbf{B}(\pi, \sigma) = \mathbf{B},\ \sigma(\mathbf{d}) = \mathbf{d}\}. \quad (3)$$

Margot [20] and Pfetsch and Rehn [30] described methods for finding the formulation symmetry group $G(\mathbf{B}, \mathbf{d}, \mathbf{c})$. Each of these methods can be used to find $G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{c})$. In [13], a more general algorithm for finding the formulation symmetry group of an MINLP is described.

Another subgroup of the symmetry group of an ILP is the symmetry group of its LP relaxation, where the two groups may or may not be the same. The subgroup property follows directly from the following definition and the definition of the symmetry group of an ILP.

Definition 2. Let $\mathcal{F}$ be the feasible set of LP (1) and

$$\begin{aligned} G^{\text{LP}} &= \{\pi \in S_n \mid \pi(\mathbf{x}) \in \mathcal{F} \text{ and } \mathbf{c}^T\pi(\mathbf{x}) = \mathbf{c}^T\mathbf{x} \;\text{ for all } \mathbf{x} \in \mathcal{F}\}, \\ G^{\text{LP}}_{\mathcal{F}} &= \{\pi \in S_n \mid \pi(\mathbf{x}) \in \mathcal{F} \;\text{ for all } \mathbf{x} \in \mathcal{F}\}, \end{aligned}$$

where S_n is the set of all permutations of indices $\{1, \dots, n\}$. Then the group G^{LP} is called the *symmetry group* of LP (1), and $G^{\text{LP}}_{\mathcal{F}}$ is called the *symmetry group of the feasible set* of LP (1).

Hence, G^{LP} of an LP is completely determined by its feasible set and its objective function. In particular, G^{LP} of an infeasible LP with n variables is S_n. Clearly, the formulation symmetry group $G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{c})$ is a subgroup of G^{LP} of LP (1), and G^{LP} of LP (1) is a subgroup of the symmetry group of ILP (1). This makes it viable to use G^{LP} or $G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{c})$ within B&B with isomorphism pruning to find a set of all non-isomorphic solutions to ILP (1). However, $G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{c})$ may be too small to reap the full benefits of B&B with isomorphism pruning. Hence, it is essential to develop methods that find G^{LP} in general.

In Section 3, we prove that G^{LP} of an LP coincides with its formulation symmetry group if the feasible set of the LP is full dimensional and it has no redundant inequalities. Therefore, the method in [20, 30] can be used to find the symmetry group of a full-dimensional LP after removing all redundant inequalities. Conversely, the formulation symmetry group of an LP can miss inherent symmetries if the LP has redundant constraints. This is discussed in Section 2.

Different LP formulations based on the same variables can have the same feasible set. When this happens we say that the two LP formulations *define* the same feasible set. We say an LP in the form of LP (1) is in *standard form* if it is feasible, has no redundant constraints, and none of the inequalities in $\mathbf{Bx} \leq \mathbf{d}$ is satisfied by every feasible $\mathbf{x}$ as an equality. Section 2 describes a method for defining the feasible set of a given feasible LP by an LP in standard form and with the same objective function. There is no known general method for finding the symmetry group of a feasible LP that is not full dimensional. In Section 3, we describe a method based on orthogonal projection matrices that finds the symmetry group of a non-full-dimensional LP in standard form.

In Section 4, we define orthogonal arrays (OAs) and describe the isomorphism and OD-equivalence operations that map OAs to OAs. In Section 5, we analytically characterize a subgroup of the LP relaxation symmetry group G^{LP} of an OA defining ILP in terms of the isomorphism and OD-equivalence operations. In Section 6, we apply the Section 3 method to compute the LP relaxation symmetry groups of many cases of an OA defining ILP formulation from [5]. There is an OA defining ILP formulation in [6] with the objective function 0 and without redundant constraints. We then make speed comparisons between using G^{LP} with this ILP formulation from [6] and two other group/formulation combinations from [10] within the B&B with isomorphism pruning algorithm from [19] for enumerating OAs up to OD-equivalence, isomorphism, and a weaker form of isomorphism. In particular, our method reduced the computation time to find all OD-equivalence classes of OA(160, 8, 2, 4) and OA(176, 8, 2, 4) by factors of 1/(2.16) and 1/(1.36) compared to the fastest known method in [6]. These are the largest 2-symbol, strength 4 cases for which classification results are available and yet only symmetry exploiting methods have successfully generated them. Moreover, for most OA defining ILPs with only inequalities that we considered, speedups gleaned from exploiting the additional LP relaxation symmetry captured by adding slack variables drastically overcome the additional computational burdens due to the added variables. In Section 7, we discuss the major findings of this paper and propose a future research project.

Throughout the paper, S_n is either the symmetric group of degree n or an isomorphic copy of it. If the action of S_n is not defined within a paragraph, then it can be assumed that S_n is the abstract symmetric group of degree n within that paragraph.

Method 1 Finding all equality constraints of a feasible LP of form (1)

1: **Input** a feasible LP L of form (1) with $m \times n$ inequality constraint matrix $\mathbf{B}$.
2: **for** $i := 1$ **to** m **step** 1 **do**
3:     **Set** $\boldsymbol{\beta}^T := \mathbf{B}_i$;   ▷ $\mathbf{B}_i$ is the ith row of $\mathbf{B}$.
4:     **Solve** LP

$$\begin{aligned} y_i \;:=\; & \min_{\mathbf{x}} \boldsymbol{\beta}^T \mathbf{x} \\ & \text{s.t. } \mathbf{A}\mathbf{x} = \mathbf{b},\ \mathbf{B}\mathbf{x} \leq \mathbf{d}; \end{aligned}$$

5: **end for**
6: **for** $i := m$ **to** 1 **step** -1 **do**
7:     **if** $y_i = d_i$ **then**   ▷ Change the ith inequality constraint of L to an equality constraint.
8:         **Append**

$$\mathbf{A} := \begin{bmatrix} \mathbf{A} \\ \boldsymbol{\beta}^T \end{bmatrix} \quad \text{and} \quad \mathbf{b} := \begin{pmatrix} \mathbf{b} \\ d_i \end{pmatrix};$$

9:         **Delete** the ith row of $\mathbf{B}$ and the ith entry of $\mathbf{d}$;
10:     **end if**
11: **end for**
12: **Output** L.

2. A method for putting a feasible LP in standard form

In this section, we provide a method for putting a feasible LP in standard form. First, we need the well-known Theorem 1. Theorem 1 leads to Method 1 for finding all equality constraints of a feasible LP.

Theorem 1. *Let $P \neq \emptyset$, $P \subseteq \mathbb{R}^n$ be the feasible set of a system of constraints*

$$\begin{aligned} \mathbf{A}\mathbf{x} &= \mathbf{b}, \\ \mathbf{B}\mathbf{x} &\leq \mathbf{d} \end{aligned} \tag{4}$$

and $\boldsymbol{\beta}_i^T$ be the ith row of $\mathbf{B}$ for $i = 1, \ldots, m$. Then P is full dimensional in the affine space $\mathbf{A}\mathbf{x} = \mathbf{b}$ if and only if there is a sequence of feasible points $\{\mathbf{x}_i\}_{i=1}^m$ of constraints (4) such that $\boldsymbol{\beta}_i^T \mathbf{x}_i < d_i$.

Remark 1. *For practical purposes, $y_i \in [d_i - 10^{-6}, d_i + 10^{-6}]$ can be used instead of $y_i = d_i$ in Step 7 of Method 1.*

It is always possible to inscribe a highly-symmetric polytope inside an asymmetric polytope so that the formulation symmetry group of the resulting system of constraints is much smaller. This idea is formalized in the following theorem. We skip the proof of this well-known result.

Theorem 2. *The formulation symmetry group of every bounded LP L with a finite number of constraints can be reduced to the identity permutation by adding redundant inequalities.*

By Theorem 2, redundant constraints can mask inherent symmetries of an LP. Hence, it is essential to remove the redundant inequalities before computing the formulation symmetry group. LPs in standard form have no redundant constraints. Method 2 finds an LP in standard form that defines the feasible set of a given feasible LP having the same objective function.

Method 2 Putting a feasible LP L of form (1) in standard form

Input a feasible LP L of form (1).
Apply Method 1 to L and overwrite L with the result;
Remove all redundant inequality constraints from L by solving a sequence of LPs;
Remove a set of all redundant equality constraints from L by using Gaussian elimination;
Output L.

3. A method for finding the symmetry group of a feasible LP

The symmetry group G^{LP} of an LP is completely determined by its feasible set and objective function. Then the symmetry group G^{LP} of a given feasible LP can be found by finding the symmetry group of an LP in standard form that has the same feasible set and objective function as the given LP. Such an LP in standard form can be obtained by applying Method 2 from Section 2. Next, we describe a method for finding the symmetry group G^{LP} of LP (1) in standard form. Let $\text{Row}(\mathbf{A})$ be the row space of $\mathbf{A}$ and

$$\mathbf{P}_{\mathbf{A}^T} = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{A} \tag{5}$$

be the orthogonal projection matrix onto $\text{Row}(\mathbf{A})$. Let p be the number of rows of $\mathbf{A}$. Thus, $p = \text{rank}(\mathbf{A})$, i.e., $\mathbf{A}$ has p linearly independent rows. For a vector $\mathbf{v} \in \mathbb{R}^p$, let $\text{diag}(\mathbf{v})$ be the diagonal matrix whose ith diagonal entry is v_i for $i \in \{1, \ldots, p\}$. Let $\boldsymbol{\sigma}$ be a vector of singular values of $\mathbf{A}$ such that $\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^T$ is a singular value decomposition of $\mathbf{A}$, where $\mathbf{U}\mathbf{U}^T = \mathbf{U}^T\mathbf{U} = \mathbf{I}_p$, $\mathbf{V}\mathbf{V}^T = \mathbf{V}^T\mathbf{V} = \mathbf{I}_n$, and the $p \times n$ matrix

$$\mathbf{D} = \begin{bmatrix}\text{diag}(\boldsymbol{\sigma}) & \mathbf{0}\end{bmatrix}$$

is based on the all zeros matrix $\mathbf{0}$ of appropriate dimension [12]. Then, equation (5) simplifies to

$$\begin{aligned}\mathbf{P}_{\mathbf{A}^T} &= \mathbf{V}\mathbf{D}^T\mathbf{U}^T(\mathbf{U}\mathbf{D}\mathbf{V}^T\mathbf{V}\mathbf{D}^T\mathbf{U}^T)^{-1}\mathbf{U}\mathbf{D}\mathbf{V}^T \\ &= \mathbf{V}\mathbf{D}^T(\mathbf{D}\mathbf{D}^T)^{-1}\mathbf{D}\mathbf{V}^T = \mathbf{V}\mathbf{I}_n^{(p)}\mathbf{V}^T,\end{aligned} \tag{6}$$

where

$$\mathbf{I}_n^{(p)} = \begin{bmatrix}\mathbf{I}_{p\times p} & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix}$$

is $n \times n$. Equation (6) should be used to compute $\mathbf{P}_{\mathbf{A}^T}$ as it does not involve matrix inversion, leading to improved accuracy especially when $\mathbf{A}$ is ill-conditioned.

Let S_n be the group of all permutations of coordinates of column vectors in $\mathbb{R}^n$. Observe that each $\pi \in S_n$ is a linear transformation from $\mathbb{R}^n$ to $\mathbb{R}^n$. Let $\boldsymbol{\Pi}$ be the matrix of $\pi \in S_n$ with respect to the standard basis $\{\mathbf{e}_1, \ldots, \mathbf{e}_n\}$. Since $(\boldsymbol{\Pi}\mathbf{v})^T = \mathbf{v}^T\boldsymbol{\Pi}^T$, right multiplication of $\mathbf{v}^T$ by $\boldsymbol{\Pi}^T$ permutes the coordinates of the row vector $\mathbf{v}^T$. The *automorphism group* of an $n \times n$ matrix $\mathbf{M}$, denoted by $G_{\mathbf{M}}$, is the set of all $\pi \in S_n$ that send $\mathbf{M}$ to itself when the rows and the columns of $\mathbf{M}$ are permuted according to π. So,

$$G_{\mathbf{M}} = \{\pi \in S_n \mid \boldsymbol{\Pi}\mathbf{M}\boldsymbol{\Pi}^T = \mathbf{M}\}.$$

For a vector space $V \subseteq \mathbb{R}^n$, define $\text{Stab}(V) = \{\pi \in S_n \mid \boldsymbol{\Pi}\mathbf{v} \in V \ \forall\ \mathbf{v} \in V\}$. Then we have the following lemma.

Lemma 1. *Let $\mathbf{A}$ be an $m \times n$ matrix with full row rank and $\mathbf{P}_{\mathbf{A}^T}$ be the orthogonal projection matrix onto Row$(\mathbf{A})$. Then* $G_{\mathbf{P}_{\mathbf{A}^T}} = \text{Stab}(\text{Row}(\mathbf{A}))$.

Proof. To prove Stab(Row($\mathbf{A}$)) $\subseteq G_{\mathbf{P}_{\mathbf{A}^T}}$, let $\pi \in$ Stab(Row($\mathbf{A}$)). Then, since $\pi \in$ Stab(Row($\mathbf{A}$)) and $\mathbf{\Pi}$ is an invertible matrix, Row($\mathbf{A}$) = Row($\mathbf{A\Pi}^T$). Hence, the set of rows of $\mathbf{A\Pi}^T$ is a basis for Row($\mathbf{A}$). Moreover, $\mathbf{\Pi}^T\mathbf{\Pi} = \mathbf{\Pi}\mathbf{\Pi}^T = \mathbf{I}$ as every permutation matrix is an orthogonal matrix. Then,

$$\mathbf{P}_{\mathbf{A}^T} = \mathbf{P}_{\mathbf{\Pi A}^T} = (\mathbf{A\Pi}^T)^T(\mathbf{A\Pi}^T(\mathbf{A\Pi}^T)^T)^{-1}\mathbf{A\Pi}^T = \mathbf{\Pi A}^T(\mathbf{A\Pi}^T\mathbf{\Pi A}^T)^{-1}\mathbf{A\Pi}^T.$$

Hence,

$$\mathbf{P}_{\mathbf{A}^T} = \mathbf{P}_{\mathbf{\Pi A}^T} = \mathbf{\Pi A}^T(\mathbf{AA}^T)^{-1}\mathbf{A\Pi}^T = \mathbf{\Pi P}_{\mathbf{A}^T}\mathbf{\Pi}^T,$$

and $\pi \in G_{\mathbf{P}_{\mathbf{A}^T}}$.

To prove $G_{\mathbf{P}_{\mathbf{A}^T}} \subseteq$ Stab(Row($\mathbf{A}$)), let $\pi \in G_{\mathbf{P}_{\mathbf{A}^T}}$. Then

$$\begin{aligned} \mathbf{P}_{\mathbf{A}^T} &= \mathbf{\Pi P}_{\mathbf{A}^T}\mathbf{\Pi}^T \\ \mathbf{P}_{\mathbf{A}^T}\mathbf{\Pi} &= \mathbf{\Pi P}_{\mathbf{A}^T}. \end{aligned} \tag{7}$$

Let Col($\mathbf{M}$) of a matrix $\mathbf{M}$ be the column space of $\mathbf{M}$ and $\mathbf{w} \in$ Row($\mathbf{A}$), where $\mathbf{w}$ is written as a column vector. Then $\mathbf{w} = \mathbf{P}_{\mathbf{A}^T}\mathbf{w}$, and by (7), we have

$$\mathbf{\Pi w} = \mathbf{\Pi P}_{\mathbf{A}^T}\mathbf{w} = \mathbf{P}_{\mathbf{A}^T}\mathbf{\Pi w}.$$

Hence, $\mathbf{\Pi w} \in$ Col($\mathbf{P}_{\mathbf{A}^T}$) = Col($\mathbf{A}^T$) = Row($\mathbf{A}$), and $\pi \in$ Stab(Row($\mathbf{A}$)). □

Let $G(\mathbf{B}, \mathbf{d}, \mathbf{c})$ be the formulation symmetry group as defined in (3) and

$$G(\mathbf{B}, \mathbf{d}, \mathbf{c}) = \{\pi \in S_n \mid \pi(\mathbf{c}) = \mathbf{c}\}$$

when $\mathbf{B}$ is the empty matrix and $\mathbf{d}$ is the empty vector. Let $G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$ be the largest subgroup of $G_{\mathbf{P}_{\mathbf{A}^T}}$ that preserves $\mathbf{c}$ and the set of inequalities in $\mathbf{Bx} \leq \mathbf{d}$. Then by Lemma 1,

$$G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})} = G_{\mathbf{P}_{\mathbf{A}^T}} \bigcap G(\mathbf{B}, \mathbf{d}, \mathbf{c}) = \text{Stab}(\text{Row}(\mathbf{A})) \bigcap G(\mathbf{B}, \mathbf{d}, \mathbf{c}). \tag{8}$$

An *automorphism* of a vertex colored, edge colored graph is a permutation of its vertices that maps adjacent vertices to adjacent vertices and preserves vertex and edge colors. The set of all such permutations forms a group called the *automorphism group* of the graph. Method 3 computes $G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$ as the intersection of the automorphism group of a vertex colored, edge colored graph and $G(\mathbf{B}, \mathbf{d}, \mathbf{c})$. The formulation symmetry group $G(\mathbf{B}, \mathbf{d}, \mathbf{c})$ can be computed as the automorphism group of a vertex colored, edge colored graph with $n + m'$ vertices, where $\mathbf{B}$ is $m' \times n$ [20, 30]. Edge coloring in this computation is necessary only if the number of distinct numerical values in the matrix $\mathbf{B}$ is larger than two [20].

Given two graphs, the *graph isomorphism* (GI) problem asks whether one can be obtained from the other by permuting vertices. Finding the generators of the automorphism group of a graph is known to be equivalent to the GI problem [15]. Finding the generators of the intersection of two groups is also equivalent to the GI problem [14]. It is easy to see that the GI problem is in NP. On the other hand, it is not known whether the GI problem is NP complete. It is also not known if the GI problem is in P. All the known algorithms for the GI problem have exponential worst-case running times. For algebraic techniques that compute the generators for the automorphism group of a graph, see [23, 26], and for the intersection of two groups, see [31].

There is available software that can be used in implementing Method 3. In Step 10, edge coloring can be implemented by using a vertex colored graph with $n\lceil\log_2(nce + 1)\rceil$ vertices, where nce is the number of distinct numerical values in $\mathbf{P}_{\mathbf{A}^T}$ and n is the number of columns of the square matrix $\mathbf{P}_{\mathbf{A}^T}$ [25]. The subgroup $H_{\mathbf{P}_{\mathbf{A}^T}}$ of the automorphism group $G_{\mathbf{P}_{\mathbf{A}^T}}$ of $\mathbf{P}_{\mathbf{A}^T}$ that preserves $\mathbf{c}$ in Step 24 and the formulation symmetry group $G(\mathbf{B}, \mathbf{d}, \mathbf{c})$ in Step 25 can both be computed as the automorphism groups of their corresponding vertex colored, edge colored graphs by using `nauty` [25, 26]. In Step 26, the intersection can be computed by using `GAP` [9].

Method 3 Computing $G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$ of an LP L of form (1) in standard form

1: **Input** $\mathbf{A}, \mathbf{B}, \mathbf{d}, \mathbf{c}$ from an LP L of form (1) in standard form.
2: **Compute** a singular value decomposition $\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^T$ and $\mathbf{P}_{\mathbf{A}^T} = \mathbf{V}\mathbf{I}_n^{(p)}\mathbf{V}^T$;
3: **Label** each distinct numerical value in $\mathbf{P}_{\mathbf{A}^T}$ with a distinct color;
4: **Set** $nce :=$ number of distinct colors in Step 3;
5: **Initialize** $\mathcal{G}(\mathbf{P}_{\mathbf{A}^T})$ to be the graph with n vertices and no edges;
6: **for** $i := 1$ **to** $(n-1)$ **step** 1 **do**
7:     **for** $j := (i+1)$ **to** n **step** 1 **do**
8:         **for** $\ell := 1$ **to** nce **step** 1 **do**
9:             **if** the (i, j)th entry of $\mathbf{P}_{\mathbf{A}^T}$ is labeled with color ℓ **then**
10:                 **Put** an edge between ith and jth vertices of $\mathcal{G}(\mathbf{P}_{\mathbf{A}^T})$ with color ℓ;
11:             **end if**
12:         **end for**
13:     **end for**
14: **end for**
15: **Label** each distinct numerical value in $\mathbf{c}$ with a distinct color;
16: **Set** $ncv :=$ number of distinct colors in Step 15;
17: **for** $i := 1$ **to** n **step** 1 **do**
18:     **for** $\ell := 1$ **to** ncv **step** 1 **do**
19:         **if** the ith entry of $\mathbf{c}$ is labeled with color ℓ **then**
20:             **Color** vertex i of $\mathcal{G}(\mathbf{P}_{\mathbf{A}^T})$ with color ℓ;
21:         **end if**
22:     **end for**
23: **end for**
24: **Compute** the automorphism group $H_{\mathbf{P}_{\mathbf{A}^T}}$ of the vertex colored, edge colored graph $\mathcal{G}(\mathbf{P}_{\mathbf{A}^T})$;
25: **Compute** $G(\mathbf{B}, \mathbf{d}, \mathbf{c})$ by computing the automorphism group of a graph [20, 30];
26: **Compute** $G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})} := H_{\mathbf{P}_{\mathbf{A}^T}} \bigcap G(\mathbf{B}, \mathbf{d}, \mathbf{c})$;
27: **Output** $G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$.

Definition 3. For an LP L of form (1) in standard form, $G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{c})^{\text{LP}}$ is defined to be the largest subgroup of G^{LP} of L that preserves the vector $\mathbf{c}$.

Lemma 2. *For an LP L of form (1) in standard form, $G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{c})^{\text{LP}} \leq G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$.*

Proof. By the definition of standard form, L is feasible. Let $\pi \in G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{c})^{\text{LP}}$ and $\mathbf{A}$ be an $m \times n$ matrix. Then each such π must preserve the facets $\mathbf{B}\mathbf{x} \leq \mathbf{d}$ and the objective function coefficient vector $\mathbf{c}$. A linear transformation preserves $\text{Row}(\mathbf{A})$ if and only if it preserves $\text{Null}(\mathbf{A}) = \text{Row}(\mathbf{A})^{\perp}$. Hence, by equations (8), it suffices to prove that π preserves $\text{Null}(\mathbf{A})$. Since the feasible set of L is a full-dimensional polytope in an affine space of dimension $n - m$, the feasible set of L contains $n-m+1$ affinely independent points $\mathbf{x}_j \in \mathbb{R}^n$ for $j = 0, \ldots, n-m$. Then the vectors $\mathbf{v}_j = \mathbf{x}_j - \mathbf{x}_0 \in \text{Null}(\mathbf{A})$ for $j = 1, \ldots, n-m$ are linearly independent. Moreover,

$$\text{Span}(\mathbf{v}_1, \ldots, \mathbf{v}_{n-m}) = \text{Null}(\mathbf{A})$$

as $\dim(\text{Null}(\mathbf{A})) = n - m$ by the rank-nullity theorem [12], where $\dim(V)$ denotes the dimension of a vector space V.

Since π is a linear transformation from $\mathbb{R}^n$ to $\mathbb{R}^n$,

$$\pi(\text{Span}(\mathbf{v}_1, \ldots, \mathbf{v}_{n-m})) = \text{Span}(\pi(\mathbf{v}_1), \ldots, \pi(\mathbf{v}_{n-m})) = \pi(\text{Null}(\mathbf{A})), \tag{9}$$

where for a vector space $V \subseteq \mathbb{R}^n$ and a linear transformation T from $\mathbb{R}^n$ to $\mathbb{R}^n$

$$T(V) = \{\mathbf{w} \in \mathbb{R}^n \mid \mathbf{w} = T(\mathbf{v}) \text{ for some } \mathbf{v} \in V\}.$$

Observe that $\pi(\mathbf{v}_j) = \pi(\mathbf{x}_j) - \pi(\mathbf{x}_0)$ and $\mathbf{A}\pi(\mathbf{v}_j) = \mathbf{A}\pi(\mathbf{x}_j) - \mathbf{A}\pi(\mathbf{x}_0) = \mathbf{b} - \mathbf{b} = \mathbf{0}$. Then, $\pi(\mathbf{v}_j) \in \text{Null}(\mathbf{A})$ for $j = 1, \ldots, n-m$ and by equations (9)

$$\pi(\text{Null}(\mathbf{A})) \subseteq \text{Null}(\mathbf{A}).$$

Since π is an invertible linear transformation and $\mathbf{v}_1, \ldots, \mathbf{v}_{n-m}$ are linearly independent, the vectors $\pi(\mathbf{v}_1), \ldots, \pi(\mathbf{v}_{n-m})$ are linearly independent. Consequently,

$$\dim(\pi(\text{Null}(\mathbf{A}))) = \dim(\text{Span}(\pi(\mathbf{v}_1), \ldots, \pi(\mathbf{v}_{n-m}))) = n - m.$$

Hence, since $\dim(\text{Null}(\mathbf{A})) = n - m$,

$$\pi(\text{Null}(\mathbf{A})) = \text{Null}(\mathbf{A}).$$

□

Let $H \leq G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$, $\mathcal{B} = \{\mathbf{e}_1, \ldots, \mathbf{e}_n\}$ be the standard basis of $\mathbb{R}^n$, and $O_1, \ldots, O_r$ be the orbits in $\mathcal{B}$ under the action of H, i.e., for each $\mathbf{x}, \mathbf{y} \in \mathcal{B}$ there exists $g \in G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$ such that $\mathbf{x} = g(\mathbf{y})$ if and only if $\mathbf{x}, \mathbf{y} \in O_i$ for some i. The fixed subspace of $\mathbb{R}^n$ under the action of H is defined as

$$\text{Fix}_H(\mathbb{R}^n) := \{\mathbf{x} \in \mathbb{R}^n \mid \gamma\mathbf{x} = \mathbf{x} \text{ for all } \gamma \in H\}.$$

Lemma 3 in [3] implies that

$$\text{Fix}_H(\mathbb{R}^n) = \text{Span}(\beta(O_1), \ldots, \beta(O_r)),$$

where for a set S of vectors

$$\beta(S) = \frac{\sum_{\mathbf{v} \in S} \mathbf{v}}{|S|}. \tag{10}$$

Let $\mathbf{E}$ be the orthogonal projection matrix onto $\text{Span}(\beta(O_1), \ldots, \beta(O_r))$ with respect to $\mathcal{B}$. Then

$$E_{ij} = \begin{cases} \frac{1}{|O_{i,j}|} & \text{if } i \text{ and } j \text{ belong to the same orbit } O_{i,j} \in \{O_1, \ldots, O_r\}, \\ 0 & \text{otherwise.} \end{cases} \tag{11}$$

The matrix $\mathbf{E}$ uniquely identifies $\text{Fix}_H(\mathbb{R}^n)$. Let L be an LP of form (1) in standard form, $\mathcal{F}^L$ be its feasible set, and $\mathcal{T}^L_{\text{Fix}_H} = \mathcal{F}^L \cap \text{Fix}_H(\mathbb{R}^n)$. Then

$$\mathcal{T}^L_{\text{Fix}_H} = \{\mathbf{x} \in \mathbb{R}^n \mid (\mathbf{I} - \mathbf{E})\,\mathbf{x} = \mathbf{0} \text{ and } \mathbf{x} \text{ is a feasible point of } L\}.$$

Now, we have the following theorem.

Theorem 3. *Let L be an LP of form (1) in standard form and $H \leq G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$. Then $\mathcal{T}^L_{\text{Fix}_H}$ is non-empty if and only if $H \leq G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{c})^{\text{LP}}$.*

Proof. By the definition of standard form, L is feasible. Let $\mathbf{v}_0 \in \mathcal{T}^L_{\text{Fix}_H}$, let

$$\mathcal{F}_{\mathbf{Bx} \leq \mathbf{d}} = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{Bx} \leq \mathbf{d}\},$$

and for a set $\mathcal{S} \subseteq \mathbb{R}^n$ and a vector $\mathbf{u} \in \mathbb{R}^n$, let $\mathcal{S} + \mathbf{u} = \{\mathbf{v} \in \mathbb{R}^n \mid \mathbf{v} = \mathbf{s} + \mathbf{u} \text{ for some } \mathbf{s} \in \mathcal{S}\}$. Since $\mathbf{v}_0$ is in the feasible set of L, $\mathbf{x} \in \mathbb{R}^n$ is in the feasible set of L if and only if

$$\mathbf{x} = \mathbf{v}_0 + \mathbf{v}$$

for some $\mathbf{v} \in \text{Null}(\mathbf{A}) \bigcap (\mathcal{F}_{\mathbf{Bx}\leq\mathbf{d}} - \mathbf{v}_0)$. Let $\mathbf{x}$ be in the feasible set of L and $\pi \in H$. Each $\pi \in H \leq G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$ preserves $\mathbf{c}$ and $\mathcal{F}_{\mathbf{Bx}\leq\mathbf{d}} - \mathbf{v}_0$ as it preserves $\mathbf{c}$, $\mathcal{F}_{\mathbf{Bx}\leq\mathbf{d}}$, and $\mathbf{v}_0$. By equations (8), π preserves $\text{Row}(\mathbf{A})$. Then, $\text{Null}(\mathbf{A}) = \text{Row}(\mathbf{A})^{\perp}$ implies that π also preserves $\text{Null}(\mathbf{A})$. Hence, π preserves $\mathbf{c}$ and $\text{Null}(\mathbf{A}) \bigcap (\mathcal{F}_{\mathbf{Bx}\leq\mathbf{d}} - \mathbf{v}_0)$. Then

$$\mathbf{A}\pi(\mathbf{x}) = \mathbf{A}\pi(\mathbf{v}_0) + \mathbf{A}\pi(\mathbf{v}) = \mathbf{A}\mathbf{v}_0 = \mathbf{b}$$

and

$$\pi(\mathbf{x}) = \pi(\mathbf{v}_0) + \pi(\mathbf{v}) = \mathbf{v}_0 + \pi(\mathbf{v}) \in \mathcal{F}_{\mathbf{Bx}\leq\mathbf{d}}.$$

Hence, $\pi \in G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\text{LP}}$. This proves $H \leq G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\text{LP}}$.

To prove the converse, let $H \leq G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\text{LP}}$ and $\mathbf{x}_0$ be a feasible point in L. Let

$$O_{\mathbf{x}_0} = \{\mathbf{y} \in \mathbb{R}^n \mid \mathbf{y} = h(\mathbf{x}_0) \text{ for some } h \in H\}$$

be the orbit of $\mathbf{x}_0$ under the action of H on $\mathbb{R}^n$ and β be the orthogonal projection operator onto $\text{Fix}_H(\mathbb{R}^n)$ as defined in equation (10). Now, since $\beta(O_{\mathbf{x}_0})$ is a convex combination of feasible points of L, $\beta(O_{\mathbf{x}_0})$ is feasible. Hence, $\beta(O_{\mathbf{x}_0}) \in \mathcal{T}^L_{\text{Fix}_H}$. □

Corollary 1. *Let L be an LP of form (1) in standard form. Then $\mathcal{T}^L_{\text{Fix}_{G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}}}$ is non-empty if and only if $G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})} = G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\text{LP}}$.*

Proof. The result follows from Lemma 2 and Theorem 3. □

Method 4 uses the formulation symmetry group $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})$ defined in equation (2) and the output $G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$ from Method 3 to find the $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\text{LP}}$ of an LP L of form (1) in standard form. Let $\mathbf{A}$ be $m\times n$ and $\mathbf{B}$ be $m'\times n$. Then the formulation symmetry group $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})$ can be computed as the automorphism group of a vertex colored, edge colored graph with $n+m+m'$ vertices, where edge coloring is necessary only if the number of distinct numerical values in the matrix $[\mathbf{A}\,\mathbf{B}]$ is larger than two [20, 30]. Method 4 requires finding a double coset decomposition of $G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$ with respect to its subgroup $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})$ and solving either 1 or q LPs, where q is the number of double cosets. In terms of computational complexity, it is not known whether there is a polynomial time algorithm for determining the number of double cosets in a double coset decomposition of a permutation group [11]. Moreover, the double coset membership problem (i.e., the problem of determining whether a given permutation in a permutation group is in a given double coset) is at least as difficult as the GI problem [11]. All the known algorithms for computing a double coset decomposition have exponential worst-case running times. Method 4 also requires computing the orbits in $\mathcal{B} = \{\mathbf{e}_1,\ldots,\mathbf{e}_n\}$ under the action of $G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$ in Step 5 and G_{ext} in Step 14. Given a set S of generators for a group G acting on a set Ω, the orbit ω^G of an element $\omega \in \Omega$ can be computed in $O(|S||\omega^G|)$ time [31], where

$$\omega^G = \{\omega' \in \Omega \mid \omega' = g\omega \text{ for some } g \in G\}.$$

Using this result, it is easy to see that the orbits in $\mathcal{B}$ under the action of $G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$ can be computed in $O(|S|n + n^2)$ time. All computations in Method 4 involving a group can be implemented in `GAP` [9], and the feasibility of LPs can be determined by using the primal or dual simplex algorithm implementation in `CPLEX` [7]. The following theorem validates Method 4.

Theorem 4. *The output of Method 4 is $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\text{LP}}$.*

Proof. The set $\mathcal{T}^L_{\text{Fix}_{G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}}} \neq \emptyset$ if and only if the LP in Step 7 is feasible. If $\mathcal{T}^L_{\text{Fix}_{G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}}} \neq \emptyset$, then $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\text{LP}} = G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$ by Corollary 1. If $\mathcal{T}^L_{\text{Fix}_{G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}}} = \emptyset$, then

$$G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c}) \leq G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\text{LP}} < G^{\text{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$$

Method 4 Computing $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\mathrm{LP}}$ of an LP L of form (1) in standard form

1: **Input** $\mathbf{A},\mathbf{B},\mathbf{b},\mathbf{d},\mathbf{c}$ from an LP L of form (1) in standard form.
2: **Initialize** $i := 1$;
3: **Compute** $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})$ by computing the automorphism group of a graph [20, 30];
4: **Compute** $G^{\mathrm{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$ by Method 3;
5: **Compute** the orbits in $\mathcal{B} = \{\mathbf{e}_1,\ldots,\mathbf{e}_n\}$ under $G^{\mathrm{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$; ▷ $\mathcal{B}$ is the standard basis.
6: **Set** $\mathbf{E}$ to be as in equation (11) for $H = G^{\mathrm{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$;
7: **Solve** the LP obtained by adding the constraint $(\mathbf{I}-\mathbf{E})\mathbf{x} = \mathbf{0}$ to L;
8: **if** the LP in Step 7 is feasible **then**
9:     **Set** $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\mathrm{LP}} := G^{\mathrm{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$ and **GOTO** Step 26;
10: **else** ▷ Compute a $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})$-double coset decomposition of $G^{\mathrm{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$.
11:     **Compute** q and a set $\{g_1,\ldots,g_q\}$ so that

$$G^{\mathrm{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})} = \bigcup_{j=1}^{q} G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})g_jG(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c});$$

12: **end if**
13: **Set** $G_{\mathrm{ext}} := \langle g_1,\ G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})\rangle$; ▷ The group generated by g_1 and $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})$.
14: **Compute** the orbits in $\mathcal{B} = \{\mathbf{e}_1,\ldots,\mathbf{e}_n\}$ under G_{ext};
15: **Set** $\mathbf{E}$ to be as in equation (11) for $H = G_{\mathrm{ext}}$;
16: **Solve** the LP obtained by adding the constraint $(\mathbf{I}-\mathbf{E})\mathbf{x} = \mathbf{0}$ to L;
17: **if** the LP in Step 16 is feasible **then**
18:     **Update** $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c}) := G_{\mathrm{ext}}$;
19: **end if**
20: **Increment** $i := i+1$;
21: **if** $i = q+1$ **then**
22:     **Set** $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\mathrm{LP}} := G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})$ and **GOTO** Step 26;
23: **else**
24:     **Set** $G_{\mathrm{ext}} := \langle g_i,\ G(\mathbf{A},\mathbf{b},\mathbf{B},\ \mathbf{d},\mathbf{c})\rangle$ and **GOTO** Step 14;
25: **end if**
26: **Output** $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\mathrm{LP}}$.

by Lemma 2 and Corollary 1. Let

$$G^{\mathrm{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})} = \bigcup_{i=1}^{q} G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})g_iG(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})$$

be a double coset decomposition of $G^{\mathrm{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})}$ obtained by using the subgroup $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})$. Now, as discussed in [4], either

$$(G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})g_iG(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c}))\bigcap G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\mathrm{LP}} = \emptyset$$

or

$$G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})g_iG(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c}) \subseteq G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\mathrm{LP}}.$$

Let G_{ext} be as in Step 13 or Step 24. Then the set $\mathcal{T}^{L}_{\mathrm{Fix}_{G_{\mathrm{ext}}}} \neq \emptyset$ if and only if the LP in Step 16 is feasible. If $\mathcal{T}^{L}_{\mathrm{Fix}_{G_{\mathrm{ext}}}} \neq \emptyset$, then $G_{\mathrm{ext}} \leq G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\mathrm{LP}}$ by Theorem 3. Hence, $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})$ can be updated with G_{ext}. □

Method 5 Computing G^{LP} of an LP L of form (1) in standard form

```
1: Input A, B, b, d, c from an LP L of form (1) in standard form.
2: Compute the reduced row echelon form of [A | b] using Gaussian elimination;   ▷ A ∈ R^{m×n}.
3: Substitute the expressions obtained in Step 2 for the basic variables in c^T x;
4: Set ĉ^T x̂ + a to be the resulting objective function from Step 3;   ▷ a ∈ R, x̂ ∈ R^{n−m}.
5: Compute G(A, b, B, d, c)^LP and G(A, b, B, d, 0)^LP by using Method 4;
6: Compute q and a set {g_1, ..., g_q} so that
       G(A, b, B, d, 0)^LP = ⋃_{i=1}^{q} G(A, b, B, d, c)^LP g_i G(A, b, B, d, c)^LP;
7: Set G^LP := G(A, b, B, d, c)^LP;
8: for i := 1 to q step 1 do
9:     Substitute the expressions obtained in Step 2 for the basic variables in c^T g_i(x);
10:    Set ĉ_i^T x̂ + a_i to be the resulting objective function from Step 9;   ▷ a_i ∈ R, x̂ ∈ R^{n−m}.
11:    if ĉ^T = ĉ_i^T and a = a_i then
12:        Update G^LP := ⟨g_i, G^LP⟩;
13:    end if
14: end for
15: Output G^LP.
```

Method 5 finds the G^{LP} of an LP L of form (1) in standard form by using the subgroup $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\text{LP}}$ of G^{LP} and the symmetry group $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{0})^{\text{LP}}$ of the feasible set of L. Both $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\text{LP}}$ and $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{0})^{\text{LP}}$ can be found by using Method 4. Method 5 requires computing a double coset decomposition of $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{0})^{\text{LP}}$ with respect to $G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\text{LP}}$. This can be done by using `GAP` [9]. The following lemma is used in proving the theorem that establishes the viability of Method 5.

Lemma 3. *Let* $\mathbf{A}$ *be an* $m \times n$ *matrix such that* $m \le n$ *and* $\text{rank}(\mathbf{A}) = m$. *Let* $\mathbf{u}_i \in \text{Null}(\mathbf{A})$ *for* $i = 1, \ldots, r$ *be linearly independent, where* $r \le n - m$. *Let* ℓ *be the set of indices of the basic variables in the reduced row echelon form of* $[\mathbf{A}\,|\,\mathbf{0}]$. *For* $i = 1, \ldots, r$, *let* $\hat{\mathbf{u}}_i \in \mathbb{R}^{n-m}$ *be obtained from* $\mathbf{u}_i$ *by deleting its entries whose indices are in* ℓ. *Then the vectors in* $\{\hat{\mathbf{u}}_1, \ldots, \hat{\mathbf{u}}_r\}$ *are linearly independent.*

Proof. It suffices to show that the columns of $[\hat{\mathbf{u}}_1\, \hat{\mathbf{u}}_2 \cdots \hat{\mathbf{u}}_r]$ are linearly independent, i.e.,

$$\text{rank}([\hat{\mathbf{u}}_1\, \hat{\mathbf{u}}_2 \cdots \hat{\mathbf{u}}_r]) = r.$$

Since $\mathbf{u}_i \in \text{Null}(\mathbf{A})$, each entry in $\mathbf{u}_i$ whose index is in ℓ is a linear combination of the entries whose indices are in $\{1, \ldots, n-m\}\backslash\ell$. Then $[\mathbf{u}_1\, \mathbf{u}_2 \cdots \mathbf{u}_r]$ is row equivalent to $[\widetilde{\mathbf{u}}_1\, \widetilde{\mathbf{u}}_2 \cdots \widetilde{\mathbf{u}}_r]$, where $\widetilde{\mathbf{u}}_i$ is obtained from $\mathbf{u}_i$ by replacing each of its entries whose index is in ℓ with 0. Now, since $\text{rank}([\mathbf{u}_1\, \mathbf{u}_2 \cdots \mathbf{u}_r]) = \text{rank}([\widetilde{\mathbf{u}}_1\, \widetilde{\mathbf{u}}_2 \cdots \widetilde{\mathbf{u}}_r])$ and $\text{rank}([\widetilde{\mathbf{u}}_1\, \widetilde{\mathbf{u}}_2 \cdots \widetilde{\mathbf{u}}_r]) = \text{rank}([\hat{\mathbf{u}}_1\, \hat{\mathbf{u}}_2 \cdots \hat{\mathbf{u}}_r])$, we get $\text{rank}([\mathbf{u}_1\, \mathbf{u}_2 \cdots \mathbf{u}_r]) = \text{rank}([\hat{\mathbf{u}}_1\, \hat{\mathbf{u}}_2 \cdots \hat{\mathbf{u}}_r])$. Hence,

$$r = \text{rank}([\mathbf{u}_1\, \mathbf{u}_2 \cdots \mathbf{u}_r]) = \text{rank}([\hat{\mathbf{u}}_1\, \hat{\mathbf{u}}_2 \cdots \hat{\mathbf{u}}_r])$$

as the vectors in $\{\mathbf{u}_1, \ldots, \mathbf{u}_r\}$ are linearly independent. □

The following theorem validates Method 5.

Theorem 5. *The output of Method 5 is the* G^{LP} *of* L.

Proof. Let the matrix $\mathbf{A}$ be $m \times n$ with $\text{rank}(\mathbf{A}) = m$. Throughout the proof, for a feasible point $\mathbf{v}$ of L, let $\hat{\mathbf{v}}$ be obtained from $\mathbf{v}$ by deleting its entries whose indices are the same as those of the basic variables in the reduced row echelon form of $[\mathbf{A} \,|\, \mathbf{b}]$, equivalently of $[\mathbf{A} \,|\, \mathbf{0}]$. Since the feasible set of L is a full-dimensional polytope in an affine space of dimension $n - m$, the feasible set of L contains affinely independent points $\mathbf{v}_j \in \mathbb{R}^n$ for $j = 0, \ldots, n-m$. Then the vectors $\mathbf{v}_j - \mathbf{v}_0 \in \text{Null}(\mathbf{A})$ for $j = 1, \ldots, n-m$ are linearly independent. Hence, the vectors in $\{\hat{\mathbf{v}}_1 - \hat{\mathbf{v}}_0, \ldots, \hat{\mathbf{v}}_{n-m} - \hat{\mathbf{v}}_0\}$ are also linearly independent by Lemma 3.

Let g_i be as in Step 6 of Method 5. Let $\hat{\mathbf{c}}^T\hat{\mathbf{x}} + a$ and $\hat{\mathbf{c}}_i^T\hat{\mathbf{x}} + a_i$ be as in Step 4 and Step 10 of Method 5. First, we have $G^{\text{LP}} \leq G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{0})^{\text{LP}}$ as $G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{0})^{\text{LP}}$ is the same as the symmetry group of the feasible set of L. Then by Definition 3,

$$G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{c})^{\text{LP}} \leq G^{\text{LP}} \leq G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{0})^{\text{LP}}.$$

Since either

$$G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{c})^{\text{LP}} g_i G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{c})^{\text{LP}} \cap G^{\text{LP}} = \emptyset$$

or

$$G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{c})^{\text{LP}} g_i G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{c})^{\text{LP}} \subseteq G^{\text{LP}},$$

it suffices to prove that $g_i \in G^{\text{LP}}$ if and only if $\hat{\mathbf{c}} = \hat{\mathbf{c}}_i$ and $a = a_i$.

Assume $g_i \in G^{\text{LP}}$. Then for $j = 0, \ldots, n-m$, we have

$$\mathbf{c}^T\mathbf{v}_j = \hat{\mathbf{c}}^T\hat{\mathbf{v}}_j + a, \quad \mathbf{c}^T g_i(\mathbf{v}_j) = \hat{\mathbf{c}}_i^T\hat{\mathbf{v}}_j + a_i, \quad \mathbf{c}^T\mathbf{v}_j = \mathbf{c}^T g_i(\mathbf{v}_j),$$

and we get

$$\mathbf{c}^T\mathbf{v}_j - \mathbf{c}^T\mathbf{v}_0 = \mathbf{c}^T g_i(\mathbf{v}_j) - \mathbf{c}^T g_i(\mathbf{v}_0).$$

Hence,

$$\hat{\mathbf{c}}^T\hat{\mathbf{v}}_j + a - \hat{\mathbf{c}}^T\hat{\mathbf{v}}_0 - a = \hat{\mathbf{c}}_i^T\hat{\mathbf{v}}_j + a_i - \hat{\mathbf{c}}_i^T\hat{\mathbf{v}}_0 - a_i,$$

and consequently,

$$\hat{\mathbf{c}}^T(\hat{\mathbf{v}}_j - \hat{\mathbf{v}}_0) = \hat{\mathbf{c}}_i^T(\hat{\mathbf{v}}_j - \hat{\mathbf{v}}_0).$$

Then,

$$(\hat{\mathbf{c}} - \hat{\mathbf{c}}_i)^T(\hat{\mathbf{v}}_j - \hat{\mathbf{v}}_0) = (\hat{\mathbf{v}}_j - \hat{\mathbf{v}}_0)^T(\hat{\mathbf{c}} - \hat{\mathbf{c}}_i) = 0 \tag{12}$$

for $j = 1, \ldots, n-m$. Equations (12) imply

$$[(\hat{\mathbf{v}}_1 - \hat{\mathbf{v}}_0)\,(\hat{\mathbf{v}}_2 - \hat{\mathbf{v}}_0)\,\cdots\,(\hat{\mathbf{v}}_{n-m} - \hat{\mathbf{v}}_0)]^T(\hat{\mathbf{c}} - \hat{\mathbf{c}}_i) = \mathbf{0}.$$

Now, the linear independence of $\hat{\mathbf{v}}_j - \hat{\mathbf{v}}_0$ for $j = 1, \ldots, n-m$ implies that the square matrix

$$[(\hat{\mathbf{v}}_1 - \hat{\mathbf{v}}_0)\,(\hat{\mathbf{v}}_2 - \hat{\mathbf{v}}_0)\,\cdots\,(\hat{\mathbf{v}}_{n-m} - \hat{\mathbf{v}}_0)]^T$$

is invertible. So, we conclude that $\hat{\mathbf{c}} - \hat{\mathbf{c}}_i = \mathbf{0}$ and $\hat{\mathbf{c}} = \hat{\mathbf{c}}_i$. Moreover, since

$$\hat{\mathbf{c}}^T\hat{\mathbf{v}}_0 + a = \mathbf{c}^T\mathbf{v}_0 = \mathbf{c}^T g_i(\mathbf{v}_0) = \hat{\mathbf{c}}_i^T\hat{\mathbf{v}}_0 + a_i,$$

we get $a = a_i$.

To prove the converse, assume $\hat{\mathbf{c}} = \hat{\mathbf{c}}_i$ and $a = a_i$. Since $g_i \in G(\mathbf{A}, \mathbf{b}, \mathbf{B}, \mathbf{d}, \mathbf{0})^{\text{LP}}$, g_i sends feasible points of L to feasible points. Now, $\hat{\mathbf{c}} = \hat{\mathbf{c}}_i$ and $a = a_i$ implies that

$$\mathbf{c}^T\mathbf{v} = \hat{\mathbf{c}}^T\hat{\mathbf{v}} + a = \hat{\mathbf{c}}_i^T\hat{\mathbf{v}} + a_i = \mathbf{c}^T g_i(\mathbf{v})$$

for each feasible point $\mathbf{v}$ of L. Hence, $g_i \in G^{\text{LP}}$ as g_i preserves the feasibility and the objective function value of each feasible point. □

Corollary 2. *The symmetry group G^{LP} of an LP L coincides with its formulation symmetry group if the feasible set of L is non-empty, full dimensional, and L has no redundant inequalities.*

Proof. Since the feasible set of L is non-empty and full dimensional, there exists no equality constraint satisfied by each feasible point of L. Then, L has no redundant inequality constraints implies that L is in standard form. WLOG assume that L has the form of LP (1). Then,

$$G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\mathrm{LP}} = G^{\mathrm{Null}}_{(\mathbf{A},\mathbf{B},\mathbf{d},\mathbf{c})} = G(\mathbf{B},\mathbf{d},\mathbf{c}) = G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c}).$$

Let L be the input to Method 5. Then $\hat{\mathbf{c}}^T\mathbf{x}+a$ becomes $\mathbf{c}^T\mathbf{x}$ in Step 10, $\hat{\mathbf{c}}_i^T\mathbf{x}+a_i$ becomes $\mathbf{c}^T g_i(\mathbf{x}) = g_i(\mathbf{c})^T\mathbf{x}$ in Step 10, and the check in Step 11 becomes $\mathbf{c}^T = (g_i(\mathbf{c}))^T$. So, the check in Step 11 requires that each new g_i to be added to $G(\mathbf{B},\mathbf{d},\mathbf{c})$ must preserve the vector $\mathbf{c}$. Then the output G^{LP} must also preserve $\mathbf{c}$. Hence, by Definition 3, $G^{\mathrm{LP}} = G(\mathbf{A},\mathbf{b},\mathbf{B},\mathbf{d},\mathbf{c})^{\mathrm{LP}} = G(\mathbf{B},\mathbf{d},\mathbf{c})$. □

4. Orthogonal arrays and their symmetries

We first define orthogonal arrays (OAs).

Definition 4. An $\mathrm{OA}(N,k,s,t)$ of strength $t \in \{0,\dots,k\}$ is an $N \times k$ array of symbols from the set $\{l_0,\dots,l_{s-1}\}$ such that each of the s^t t-tuples from $\{l_0,\dots,l_{s-1}\}^t$ appears $\lambda = N/s^t$ times in every $N \times t$ subarray.

By Definition 4, every $N \times k$ array with symbols from a set $\{l_0,\dots,l_{s-1}\}$ is an $\mathrm{OA}(N,k,s,0)$ and vice versa. For fixed N, k, s, and $t \in \{0,\dots,k\}$, $\mathrm{OA}(N,k,s,t)$ have many inherent symmetries, where each symmetry is a bijective map from the set of all $\mathrm{OA}(N,k,s,0)$ to the set of all $\mathrm{OA}(N,k,s,0)$ that preserves the $\mathrm{OA}(N,k,s,t)$ property. In particular, each row permutation is a symmetry of $\mathrm{OA}(N,k,s,t)$ for all $t \in \{0,\dots,k\}$. We call each such symmetry a *trivial symmetry* of OAs.

A major source of non-trivial symmetries of $\mathrm{OA}(N,k,s,t)$ for $t \in \{0,\dots,k-1\}$ is the set of isomorphism operations [6]. (For $t = k$, it is easy to show that every isomorphism operation is a trivial symmetry.) Next, we define isomorphism operations and the group of isomorphism operations that act on $\mathrm{OA}(N,k,s,t)$ for $t \in \{0,\dots,k\}$.

Definition 5. Each of the $k!(s!)^k$ operations that involve permuting columns and the elements of $\{l_0,\dots,l_{s-1}\}$ within each column of an $N \times k$ array with symbols from $\{l_0,\dots,l_{s-1}\}$ is called an *isomorphism operation.* The set of all isomorphism operations forms a group called the *paratopism group* [8].

We denote the paratopism group acting on $\mathrm{OA}(N,k,s,t)$ with $G^{\mathrm{iso}}(k,s)$. Two $\mathrm{OA}(N,k,s,t)$s $\mathbf{X}$ and $\mathbf{Y}$ are *isomorphic* if

$$\text{the set of rows of } \mathbf{X} = \text{the set of rows of } g(\mathbf{Y})$$

for some $g \in G^{\mathrm{iso}}(k,s)$ [35]. It is well known that $G^{\mathrm{iso}}(k,s) \cong S_s \wr S_k$ [8], where $S_s \wr S_k$ is the wreath product of the symmetric group of degree s and the symmetric group of degree k. For a definition of the wreath product of groups, see [33]. In [6], OD-equivalence of $\mathrm{OA}(N,k,2,t)$ for even t was defined and used to classify all non-isomorphic $\mathrm{OA}(160,k,2,4)$ and $\mathrm{OA}(176,k,2,4)$. To define OD-equivalence of OAs, we first need the concept of Hadamard equivalence from [22].

Definition 6. Two $N \times k$ arrays $\mathbf{Y}_1$ and $\mathbf{Y}_2$ with symbols from $\{-1,1\}$ are *Hadamard equivalent* if $\mathbf{Y}_2$ can be obtained from $\mathbf{Y}_1$ by applying a sequence of signed permutations (permutations that may or may not be followed by sign changes) to the columns or rows of $\mathbf{Y}_1$.

Definition 7. Two $N \times k$ arrays $\mathbf{X}_1$ and $\mathbf{X}_2$ with symbols from $\{-1, 1\}$ are *OD-equivalent* if $[\mathbf{1}, \mathbf{X}_1]$ and $[\mathbf{1}, \mathbf{X}_2]$ are Hadamard equivalent.

Clearly, two isomorphic OA$(N, k, 2, t)$ with symbols from $\{-1, 1\}$ are OD-equivalent. However, there exist OD-equivalent OA$(N, k, 2, t)$ that are not isomorphic [6]. In what follows, we describe the operations other than the isomorphism operations that send an OA$(N, k, 2, t)$ to one of its OD-equivalent copies. Let $\mathbf{Y}$ be an $N \times k$ array with symbols from $\{-1, 1\}$. For each $i \in \{1, \ldots, k\}$, define the column operation R_i on $\mathbf{Y}$ to be

$$\mathbf{Y} = \begin{bmatrix}\mathbf{y}_1 & \cdots & \mathbf{y}_i & \cdots & \mathbf{y}_k\end{bmatrix} \xrightarrow{R_i} \begin{bmatrix}\mathbf{y}_1 \odot \mathbf{y}_i & \cdots & \mathbf{y}_{i-1} \odot \mathbf{y}_i & \mathbf{y}_i & \mathbf{y}_{i+1} \odot \mathbf{y}_i & \cdots & \mathbf{y}_k \odot \mathbf{y}_i\end{bmatrix},$$

where

$$\mathbf{u} \odot \mathbf{v} = \begin{bmatrix} u_1 v_1 \\ \vdots \\ u_n v_n \end{bmatrix}$$

for $\mathbf{u}, \mathbf{v} \in \{-1, 1\}^n$. Now, we have the following definition.

Definition 8. Let the group generated by $R_1, \ldots, R_k$ and the elements of $G^{\text{iso}}(k, 2)$ be denoted by $G(k)^{\text{OD}}$. Each element of $G(k)^{\text{OD}}$ is called an *OD-equivalence operation*.

The proof of the next result is a modification of the proof of Theorem 2 in [1]. It also fills in the details skipped in [1]. We present Theorem 2 in [1] as Theorem 10 in Section 5.

Lemma 4. *Let* $\mathbf{Y} = [\mathbf{y}_1\, \mathbf{y}_2 \cdots \mathbf{y}_k] \in \{-1, 1\}^{N \times k}$*, and* $G(k)^{\text{OD}} = \langle G^{\text{iso}}(k, s), R_1, \ldots, R_k \rangle$ *act on* $\mathbf{Y}$ *as in Definition 8. Then* $G(k)^{\text{OD}} \cong S_2^k \rtimes S_{k+1}$*.*

Proof. Let $R = \langle R_1, \ldots, R_k \rangle \leq G(k)^{\text{OD}}$ and $R_0 = e$ be the identity element of $G(k)^{\text{OD}}$. Let $\hat{S}_k$ be the group of all permutations that permute the columns of $\mathbf{Y}$. Since $R_i R_j R_i$ permutes $\mathbf{y}_i$ and $\mathbf{y}_j$, we have $\hat{S}_k \leq G(k)^{\text{OD}}$ and $R_i R_j R_i \hat{S}_k = \hat{S}_k$. Then, $R_i R_j R_i \hat{S}_k = \hat{S}_k$ implies $R_j R_i \hat{S}_k = R_i \hat{S}_k$ as $R_i^2 = e$. Since $R_j R_i \hat{S}_k = R_i \hat{S}_k$ for each distinct $i, j \in \{1, \ldots, k\}$, there are $k + 1$ left cosets of $\hat{S}_k$ in R. So each element $x \in R$ can be written uniquely as $x = R_j \tau$ for some $\tau \in \hat{S}_k$ and $j \in \{0, \ldots, k\}$. Consequently, $|R| = (k + 1)!$. Then, $|R| < \infty$ implies that every element of R can be written as a finite product of the R_is. Let

$$\sigma_{ij} = R_i R_j R_i \in \hat{S}_k \tag{13}$$

for each distinct $i, j \in \{1, \ldots, k\}$. Then, for each distinct $i_1, \ldots, i_r \in \{1, \ldots, k\}$,

$$R_{i_1} R_{i_2} \cdots R_{i_{r-1}} R_{i_r} R_{i_1} = R_{i_1} R_{i_2} R_{i_1} R_{i_1} \cdots R_{i_1} R_{i_1} R_{i_{r-1}} R_{i_1} R_{i_1} R_{i_r} R_{i_1} = \sigma_{i_1 i_2} \sigma_{i_1 i_3} \cdots \sigma_{i_1 i_r} \tag{14}$$

$$R_{i_1} R_{i_2} \cdots R_{i_r} = R_{i_1} R_{i_r} R_{i_r} R_{i_2} R_{i_r} R_{i_r} \cdots R_{i_r} R_{i_r} R_{i_{r-1}} R_{i_r} = R_{i_1} R_{i_r} \sigma_{i_r i_2} \sigma_{i_r i_3} \cdots \sigma_{i_r i_{r-1}} \tag{15}$$

by equation (13) and $R_i^2 = e$. Now, since $R_{i_1} R_{i_r} = R_{i_r} \sigma_{i_1 i_r}$, equation (15) becomes

$$R_{i_1} R_{i_2} \cdots R_{i_r} = R_{i_r} \sigma_{i_r i_1} \sigma_{i_r i_2} \sigma_{i_r i_3} \cdots \sigma_{i_r i_{r-1}}. \tag{16}$$

Then, given $x = R_{j_1} R_{j_2} \cdots R_{j_p} \in R$ for some not necessarily distinct $j_1, \ldots, j_p \in \{1, \ldots, k\}$, x can be reduced to its unique form $x = R_j \tau$ by first applying equation (14) from right to left and then applying equation (16) (if applicable) once equation (14) can no longer be applied.

Given $x = R_{j_1} R_{j_2} \ldots R_{j_p} \in R$ for some not necessarily distinct $j_1, \ldots, j_p \in \{1, \ldots, k\}$, define $\psi : R \to S_{k+1}$ by

$$\psi(x) = \psi(R_{j_1} R_{j_2} \cdots R_{j_p}) = (j_1, k + 1)(j_2, k + 1) \cdots (j_p, k + 1).$$

We now prove that ψ is an isomorphism. First, assuming that ψ is well-defined, it is clear that ψ is a homomorphism. Second, S_{k+1} is generated by the elements of $\{(1,k+1),(2,k+1),\ldots,(k,k+1)\}$ as each transposition (i,j) satisfies

$$(i,j) = (i,k+1)(j,k+1)(i,k+1). \tag{17}$$

Thus, ψ is onto S_{k+1}. Assuming that ψ is a well-defined map, injectivity of ψ follows from the facts that $|S_{k+1}| = |R|$ and ψ is onto S_{k+1}. Hence, it suffices to show that ψ is a well-defined map. Let $R'_i = (i,k+1)$ for $i \in \{1,\ldots,k\}$ and $\sigma'_{ij} = (i,j) \in S_k$ for each distinct $i,j \in \{1,\ldots,k\}$. Let $R'_0 = e'$ be the identity permutation in S_{k+1}. Then equation (17) and $R'^2_i = e'$ imply

$$R'_{i_1}R'_{i_2}\cdots R'_{i_{r-1}}R'_{i_r}R'_{i_1} = \sigma'_{i_1i_2}\sigma'_{i_1i_3}\cdots\sigma'_{i_1i_r} \tag{18}$$
$$R'_{i_1}R'_{i_2}\cdots R'_{i_r} = R'_{i_r}\sigma'_{i_ri_1}\sigma'_{i_ri_2}\sigma'_{i_ri_3}\cdots\sigma'_{i_ri_{r-1}} \tag{19}$$

the same way equation (13) and $R_i^2 = e$ imply equations (14) and (16). Let $x = R_{j_1}R_{j_2}\cdots R_{j_{p_1}} \in R$ and $y = R_{j'_1}R_{j'_2}\cdots R_{j'_{p_2}} \in R$ be such that $x = y$. To finish the proof, we need to show that $\psi(x) = \psi(y)$. The equality $x = y$ implies that $x = y = R_j\tau$ for some $j \in \{0,\ldots,k\}$ and $\tau \in \hat{S}_k$. Moreover, $x = y = R_j\tau$ can be obtained by first applying equation (14) from right to left and then applying equation (16) (if applicable) to $x = R_{j_1}R_{j_2}\cdots R_{j_{p_1}}$ and $y = R_{j'_1}R_{j'_2}\cdots R_{j'_{p_2}}$. Then $\psi(x) = \psi(y) = R'_j\tau'$, for some $\tau' \in S_k$ can be obtained by first applying equation (18) from right to left and then applying equation (19) (if applicable) to $\psi(x) = R'_{j_1}R'_{j_2}\cdots R'_{j_{p_1}}$ and $\psi(y) = R'_{j'_1}R'_{j'_2}\cdots R'_{j'_{p_2}}$. Therefore, we conclude that $R \cong S_{k+1}$.

Let $\phi \in Z_2^k$ be a multiplication of some subset of columns of $\mathbf{Y}$ by -1. Then $R_i\phi R_i = R_i^{-1}\phi R_i = \phi'$, where $\phi' \in Z_2^k$. This implies that $Z_2^k \trianglelefteq G(k)^{\mathrm{OD}}$ as $Z_2^k \trianglelefteq Z_2 \wr \hat{S}_k \leq G(k)^{\mathrm{OD}}$. Then, $Z_2^k \trianglelefteq G(k)^{\mathrm{OD}}$, $Z_2^k \cap R = \{e\}$, and $R \leq G(k)^{\mathrm{OD}}$ imply $Z_2^k \rtimes R \leq G(k)^{\mathrm{OD}}$. Now, $G(k)^{\mathrm{OD}} = \langle R, Z_2^k\rangle$ and $Z_2^k \trianglelefteq G(k)^{\mathrm{OD}}$. So, for each $g \in G(k)^{\mathrm{OD}}$, $g = rw$ for some $r \in R$ and $w \in Z_2^k$. Consequently, $|G(k)^{\mathrm{OD}}| \leq |R||Z_2^k| = (k+1)!2^k$. Hence, we get $G(k)^{\mathrm{OD}} = Z_2^k \rtimes R \cong S_2^k \rtimes S_{k+1}$. □

For even t, a major source of non-trivial symmetries of $\mathrm{OA}(N,k,2,t)$ that are not isomorphism operations are the OD-equivalence operations that are not in $G^{\mathrm{iso}}(k,2)$ [6]. The following theorem shows that the OD-equivalence operations are indeed symmetries of $\mathrm{OA}(N,k,2,t)$ when t is even.

Theorem 6. *Let $\mathbf{Y}$ be an OA$(N,k,2,t)$ with symbols from $\{-1,1\}$ and strength $t \geq 1$. Then $\mathbf{X}$ is OD-equivalent to $\mathbf{Y}$ if and only if there exists an OD-equivalence operation g such that*

$$\textit{the set of rows of } \mathbf{X} = \textit{the set of rows of } g(\mathbf{Y}).$$

Moreover, if $\mathbf{X}$ is OD-equivalent to $\mathbf{Y}$, then $\mathbf{X}$ is an OA$(N,k,2,2\lfloor\frac{t}{2}\rfloor)$.

We first show that two OAs each with strength at least 1 are OD-equivalent if and only if the set of rows of one can be obtained from that of the other by applying an element of $G(k)^{\mathrm{OD}}$ to each row in the set.

Lemma 5. *Let $\mathbf{X}$, $\mathbf{Y}$ be OA$(N,k,2,t_1)$, OA$(N,k,2,t_2)$ with symbols from $\{-1,1\}$ and strengths $t_1 \geq 1$, $t_2 \geq 1$. Then $\mathbf{X}$ and $\mathbf{Y}$ are OD-equivalent if and only if*

$$\textit{the set of rows of } \mathbf{X} = \textit{the set of rows of } g(\mathbf{Y})$$

for some $g \in G(k)^{\mathrm{OD}}$.

Proof. By the definition of OD-equivalence, $\mathbf{X}$ and $\mathbf{Y}$ are OD-equivalent if and only if

$$\mathbf{\Pi D}[\mathbf{1}\,\mathbf{X}] = [\mathbf{1}\,\mathbf{Y}]\mathbf{D}_1\mathbf{\Pi}_1 \tag{20}$$

for some permutation matrices $\mathbf{\Pi}$ and $\mathbf{\Pi}_1$ and diagonal matrices $\mathbf{D}$ and $\mathbf{D}_1$ whose diagonal entries are in $\{-1,1\}$. Now, $\mathbf{Y}$ is an OA with strength at least 1; consequently, $\mathbf{\Pi D}[\mathbf{1}\,\mathbf{X}] = [\mathbf{1}\,\mathbf{Y}]\mathbf{D}_1\mathbf{\Pi}_1$ must have $\pm\mathbf{1}$ as a column exactly once. Since $\mathbf{X}$ is also an OA with strength at least 1, we must have either $\mathbf{D} = \pm\mathbf{I}$ or $\mathbf{D} = \pm\mathrm{diag}(\mathbf{x}_i)$, where $\mathbf{x}_i$ is the ith column of $\mathbf{X}$ for some i. When $\mathbf{D} = \pm\mathbf{I}$, equation (20) holds if and only if $\mathbf{X}$ is isomorphic to $\mathbf{Y}$. This is true if and only if $\mathbf{X}$ can be obtained from $\mathbf{Y}$ by applying an element g of $G^{\mathrm{iso}}(k,2) \le G(k)^{\mathrm{OD}}$. When $\mathbf{D} = \pm\mathrm{diag}(\mathbf{x}_i)$ for some i, then equation (20) holds if and only if

$$\{\mathbf{\Pi D}[\mathbf{1}\,\mathbf{X}]\}\backslash\{\mathbf{1},-\mathbf{1}\} = \{[\mathbf{1}\,\mathbf{Y}]\mathbf{D}_1\mathbf{\Pi}_1\}\backslash\{\mathbf{1},-\mathbf{1}\}, \tag{21}$$

where $\{\mathbf{D}[\mathbf{1}\,\mathbf{X}]\}\backslash\{\mathbf{1},-\mathbf{1}\} = \{\pm\mathbf{x}_i \odot \mathbf{x}_1, \ldots, \pm\mathbf{x}_i \odot \mathbf{x}_{i-1}, \pm\mathbf{x}_i, \pm\mathbf{x}_i \odot \mathbf{x}_{i+1}, \ldots, \pm\mathbf{x}_i \odot \mathbf{x}_k\}$ and $\{\mathbf{M}\}$ is the set of columns of a matrix $\mathbf{M}$. Equation (21) holds if and only if

$$\text{the set of rows of } h_1(\mathbf{X}) = \text{the set of rows of } h_2(\mathbf{Y})$$

for some $h_1, h_2 \in G(k)^{\mathrm{OD}}$. The result now follows from

$$\text{the set of rows of } \mathbf{X} = \text{the set of rows of } h_1^{-1}h_2(\mathbf{Y})$$

by taking $g = h_1^{-1}h_2$. □

Next, we prove Theorem 6, but first we need the concept of J-characteristics and the subsequent lemma from [35].

Definition 9. Let $\mathbf{Y} = [y_{ij}]$ be an $N \times k$ array with symbols from $\{-1,1\}$. Let $r \in \{1,\ldots,k\}$ and $\ell = \{i_1,\ldots,i_r\} \subseteq \{1,\ldots,k\}$. Then the integers

$$J_r(\ell)(\mathbf{Y}) := \sum_{i=1}^{N} \prod_{j\in\ell} y_{ij}$$

are called the *J-characteristics* of $\mathbf{Y}$. (For $r = 0$, $J_0(\emptyset)(\mathbf{Y}) := N$.)

Lemma 6 (Stufken and Tang [35])**.** *An $N\times k$ array $\mathbf{Y}$ with symbols from $\{-1,1\}$ is an OA$(N,k,2,t)$ if and only if $J_r(\ell)(\mathbf{Y}) = 0$ for all $\ell \subseteq \{1,\ldots,k\}$ such that $|\ell| = r$ and $r \in \{1,\ldots,t\}$.*

Lemma 7. *Let $\ell \subseteq \{1,\ldots,k\}$ be such that $|\ell| = r > 0$. Let g be an OD-equivalence operation and $g(\mathbf{Y})$ be the array obtained after g is applied to $\mathbf{Y}$. Then*

$$J_r(\ell)(g(\mathbf{Y})) = \pm J_{r'}(\ell')(\mathbf{Y})$$

for some $\ell' \subseteq \{1,\ldots,k\}$, where

$$|\ell'| = r' = \begin{cases} r \text{ or } r+1 & \text{if } r \text{ is odd,} \\ r \text{ or } r-1 & \text{otherwise.} \end{cases} \tag{22}$$

Proof. Let $i \in \{1,\ldots,k\}$ and R_i be as in Definition 8. Then,

$$J_r(\ell)(R_i(\mathbf{Y})) = \begin{cases} J_r(\ell)(\mathbf{Y}) & \text{if } r \text{ is even and } i \notin \ell, \\ J_{r-1}(\ell\backslash\{i\})(\mathbf{Y}) & \text{if } r \text{ is even and } i \in \ell, \\ J_{r+1}(\ell\cup\{i\})(\mathbf{Y}) & \text{if } r \text{ is odd and } i \notin \ell, \\ J_r(\ell)(\mathbf{Y}) & \text{if } r \text{ is odd and } i \in \ell. \end{cases} \tag{23}$$

Let $R = \langle R_1, \dots, R_k \rangle$ and Z_2^k be the group of all possible sign switches of columns of $\mathbf{Y}$. Then by the proof of Lemma 4, $g = g_1 g_2$, where $g_1 \in R$ and $g_2 \in Z_2^k$. Then by equation (23),

$$J_r(\ell)(g(\mathbf{Y})) = J_r(\ell)(g_1(g_2(\mathbf{Y}))) = J_{r'}(\ell')(g_2(\mathbf{Y}))$$

for some $\ell' \subseteq \{1, \dots, k\}$ and $r' = |\ell'|$ as in equation (22). Now, $g_2(\mathbf{Y})$ is obtained from $\mathbf{Y}$ by multiplying some columns of $\mathbf{Y}$ by -1. Hence,

$$J_r(\ell)(g(\mathbf{Y})) = J_{r'}(\ell')(g_2(\mathbf{Y})) = \pm J_{r'}(\ell')(\mathbf{Y}).$$

□

Finally, observe that Theorem 6 follows from Lemmas 5, 6, and 7.

5. The LP relaxation symmetry group of an OA(N, k, s, t) defining ILP

First, for each N, k, s, t combination, we describe the OA(N, k, s, t) defining ILP in [5] whose feasible set contains a set of all non-isomorphic (non-OD-equivalent if $s = 2$) OA(N, k, s, t). Define the *frequency vector* of an OA(N, k, s, t) to be $\mathbf{x} := (x_1, \dots, x_{s^k})$ whose $\left(\sum_{j=1}^{k} i_j s^{k-j} + 1\right)$th entry is the number of times the symbol combination $(l_{i_1}, \dots, l_{i_k}) \in \{l_0, \dots, l_{s-1}\}^k$ appears in an OA(N, k, s, t). Then $\mathbf{x}$ must be a feasible point of the ILP

$$\begin{aligned} & \min \quad \mathbf{1}^T \mathbf{x} \\ \text{s.t.} \quad & \sum_{\{i_1,\dots,i_k\} \setminus \{i_{j_1},\dots,i_{j_t}\} \in \{0,\dots,s-1\}^{k-t}} x_{[i_1 s^{k-1} + \dots + i_k s^{k-k} + 1]} = \frac{N}{s^t} \\ & \text{for each } \{j_1, \dots, j_t\} \subseteq \{1, \dots, k\} \text{ and } (i_{j_1}, \dots, i_{j_t}) \in \{0, \dots, s-1\}^t, \\ & 0 \le x_i \le p_{\max}, \quad x_i \in \mathbb{Z}, \quad \text{for } i \in \{1, \dots, s^k\} \end{aligned} \tag{24}$$

with a large formulation symmetry group, where $\mathbf{1}$ is the vector of all ones and $p_{\max} \le N/s^t$ is a positive integer computed as in [5]. All feasible points (solutions) of ILP (24) are optimal as each OA(N, k, s, t) must have N rows. So, the objective function $\mathbf{1}^T \mathbf{x}$ was introduced to formulate a constraint satisfaction problem for OAs as an ILP.

Let $G(k, s, t)$ be the formulation symmetry group of ILP (24). In [10], it is shown that $G(k, s, t) \cong S_s \wr S_k$ for $1 \le t \le k - 1$, and each element of $G(k, s, t)$ sends the frequency vector of an OA(N, k, s, t) to that of one of its isomorphic copies. Hence, for $1 \le t \le k - 1$, $G(k, s, t) = G^{\text{iso}}(k, s)$ as $G^{\text{iso}}(k, s)$'s action on the frequency vector of an OA(N, k, s, t) is identical to that of $G(k, s, t)$. In [5], all OA(N, k, s, t) for many N, k, s, t combinations were enumerated up to isomorphism by finding a set of all non-isomorphic solutions to ILP (24) under the action of $G(k, s, t)$.

Let $G(k, s, t)^{\text{LP}}$ be the LP relaxation symmetry group of ILP (24). Since the formulation symmetry group of an LP is a subgroup of the LP relaxation symmetry group, we have the following result.

Lemma 8. *The LP relaxation symmetry group $G(k, s, t)^{\text{LP}}$ contains $G(k, s, t) = G^{\text{iso}}(k, s) \cong S_s \wr S_k$, and hence $|G(k, s, t)^{\text{LP}}| \ge |S_s \wr S_k| = k!(s!)^k$.*

Having linearly dependent constraints in an ILP formulation slows down a B&B algorithm as

LP relaxations in the B&B search tree take longer to solve. The ILP formulation

$$\begin{aligned} &\min \quad 0 \\ &\text{s.t.} \sum_{\{i_1,\ldots,i_k\}\setminus\{i_{j_1},\ldots,i_{j_q}\}\in\{0,\ldots,s-1\}^{k-q}} x_{[i_1 s^{k-1}+\cdots+i_k s^{k-k}+1]} = \frac{N}{s^q} \\ &\quad \text{for each } q\in\{0,\ldots,t\},\ \{j_1,\ldots,j_q\}\subseteq\{1,\ldots,k\}, \\ &\quad \text{and } (i_{j_1},\ldots,i_{j_q})\in\{0,\ldots,s-2\}^q, \\ &\quad 1\le x_1, \quad 0\le x_i\le p_{\max}, \quad x_i\in\mathbb{Z}, \quad \text{for } i\in\{1,\ldots,s^k\} \end{aligned} \tag{25}$$

from [6] improves the ILP (24) formulation by replacing its set of linearly dependent equality constraints with a row equivalent, yet linearly independent set of equalities. (We are going to prove that ILP (25) has a linearly independent set of equality constraints.) ILP (25) has a smaller formulation symmetry group than that of ILP (24), and yet the following remark holds.

Remark 2. *The set of all lexicographically minimum solutions of ILP (25) under the action of $G^{\text{iso}}(k,s) = G(k,s,t)$ or $G(k,s,t)^{\text{LP}}$ is the same as that of ILP (24).*

By Lemma 8, we have $G^{\text{iso}}(k,s) = G(k,s,t) \le G(k,s,t)^{\text{LP}}$. Hence, it suffices to justify Remark 2 for $G^{\text{iso}}(k,s)$. Since $G^{\text{iso}}(k,s)$ acts transitively on the indices of the variables of ILP (24), all lexicographically minimum solutions of ILP (24) under $G^{\text{iso}}(k,s)$ satisfy $1 \le x_1$. Then Remark 2 follows as the equality constraints of ILP (24) and ILP (25) are row equivalent and ILP (24) and ILP (25) have the same inequalities if $1 \le x_1$ is deleted from ILP (25). Hence, finding a set of all non-isomorphic solutions to ILP (25) under the action of $G(k,s,t)$ is equivalent to classifying $\text{OA}(N,k,s,t)$ up to isomorphism.

Let $\mathbf{A}(k,s,t)$ and $\mathbf{A}'(k,s,t)$ be the equality constraint matrices of ILPs (24) and (25). Next, we are going to establish a complete characterization of $G(k,s,t)^{\text{LP}}$ that only involves the projection matrix $\mathbf{P}_{\mathbf{A}(k,s,t)^T} = \mathbf{P}_{\mathbf{A}'(k,s,t)^T}$ by proving the following theorem.

Theorem 7. *Let $p_{\max} \ne \lambda/(s^{k-t})$ in ILP (24). Then $G(k,s,t)^{\text{LP}} = G_{\mathbf{P}_{\mathbf{A}'(k,s,t)^T}}$, where $G_{\mathbf{P}_{\mathbf{A}'(k,s,t)^T}}$ is the automorphism group of $\mathbf{P}_{\mathbf{A}'(k,s,t)^T}$.*

We are also going to show

$$\left|G(k,s,t)^{\text{LP}}\right| \ge \begin{cases} (k+1)!2^k & \text{if } s=2 \text{ and } t \text{ is even,} \\ k!\,(s!)^k & \text{otherwise} \end{cases} \tag{26}$$

for $1 \le t \le k-1$ and prove the following theorem.

Theorem 8. *Let $1 \le t \le k-1$ and $G(k,s,t)^{\text{LP}}$ satisfy inequality (26) as an equality. Let $G(k,s,t)^{\text{LP}}$ be used within B&B with isomorphism pruning to find a set of all non-isomorphic solutions $\mathcal{F}$ of ILP (25). Then*

$$\mathcal{F} = \begin{cases} \textit{a set of all non-OD-equivalent OA(N,k,2,t)} & \textit{if } t \textit{ is even and } s=2, \\ \textit{a set of all non-isomorphic OA(N,k,s,t)} & \textit{otherwise.} \end{cases}$$

In [1], inequality (26) was proven to be an equality, i.e., the hypothesis of Theorem 8 was proven when $s=2$, $t=1$ and when $s=t=2$, $k\ge 4$ by proving the following two theorems.

Theorem 9 (Arquette and Bulutoglu [1])**.** $G(k,2,1)^{\text{LP}} \cong S_2^k \rtimes S_k$.

Theorem 10 (Arquette and Bulutoglu [1])**.** *For $k \ge 4$, $G(k,2,2)^{\text{LP}} \cong S_2^k \rtimes S_{k+1}$.*

Once a set of all non-OD-equivalent OA$(N, k, 2, t)$ is found, the method in [6] for extracting a set of all non-isomorphic OA$(N, k, 2, t)$ can be used to find a set of all non-isomorphic OA$(N, k, 2, t)$. The time it takes for such an extraction was observed to be insignificant compared to the time it takes to find a set of all non-OD-equivalent OA$(N, k, 2, t)$ [6].

Theorem 1 in [10] modifies ILP (24) to an ILP without equalities by deleting a set of basic variables after Gaussian elimination. Such a modification is only possible because at the end of Gaussian elimination the coefficient of each basic variable is one and the coefficient of each free variable is an integer. The deleted variables are in fact a set of slack variables of the resulting ILP's LP relaxation. For the computational experiments in [10], the objective function of this ILP was taken to be the zero function. Let $G(k, s, t)^{\mathrm{LP}\leq}$ be the LP relaxation symmetry group of the resulting ILP once the basic variables are deleted. In [10], it is shown that $G(k, s, t)^{\mathrm{LP}\leq} \cong S_{s-1} \wr S_k$. Hence,

$$\left|G(k, s, t)^{\mathrm{LP}\leq}\right| = k!((s-1)!)^k < |G(k, s, t)| = k!(s!)^k.$$

On the other hand, since ILP (24) has equality constraints, it is not clear whether $G(k, s, t) = G(k, s, t)^{\mathrm{LP}}$.

Next, we describe yet another OA$(N, k, 2, t)$ defining ILP formulation developed in [1]. We will add redundant equalities to this formulation to show that $G(k, 2, t)^{\mathrm{LP}}$ contains $G(k)^{\mathrm{OD}}$ when t is even. This ILP has the same variables, the same inequalities (excluding the inequality $1 \leq x_1$ in ILP (25)) as in ILPs (24) and (25). Moreover its equality constraints are row equivalent to those of ILPs (24) and (25). Hence, its LP relaxation's feasible set is the same as that of the LP relaxation of ILP (24).

Let the transpose of row vectors of $\mathbf{Z} = [\mathbf{z}_1\, \mathbf{z}_2\, \cdots\, \mathbf{z}_k]$ be all 2^k vectors in $\{-1, 1\}^k$. For $i_1 < \cdots < i_r \in \{1, \ldots, k\}$ with $r \geq 2$, let $\mathbf{z}_{i_1,\ldots,i_r}$ be the r-way Hadamard product $\mathbf{z}_{i_1} \odot \cdots \odot \mathbf{z}_{i_r}$, where the pth entry of $\mathbf{z}_{i_1} \odot \cdots \odot \mathbf{z}_{i_r}$ is the product of the entries on the pth row of the matrix $[\mathbf{z}_{i_1}\, \mathbf{z}_{i_2}\, \cdots\, \mathbf{z}_{i_r}]$. Let x_p for $p = 1, \ldots, 2^k$ be the number of times the pth row of $\mathbf{Z}$ appears in an OA$(N, k, 2, t)$. Now, by Lemma 6, the ILP

$$\begin{aligned} & \min\ \mathbf{1}^T\mathbf{x} \\ \text{s.t.}\quad & \mathbf{1}^T\mathbf{x} = N, \\ & \begin{bmatrix} \mathbf{M} \\ -\mathbf{M} \end{bmatrix} \mathbf{x} = \mathbf{0}, \\ & \mathbf{x} \geq \mathbf{0},\ \mathbf{x} \in \mathbb{Z}^{2^k} \end{aligned} \tag{27}$$

is an OA$(N, k, 2, t)$ defining ILP formulation, where $\mathbf{M}$ is the $\sum_{i=1}^{t} \binom{k}{i} \times 2^k$ matrix

$$\mathbf{M} = \begin{bmatrix} \mathbf{z}_1^T \\ \vdots \\ \mathbf{z}_k^T \\ \mathbf{z}_{1,2}^T \\ \vdots \\ \mathbf{z}_{k-t+1,\ldots,k}^T \end{bmatrix}.$$

The constraints $\mathbf{1}^T\mathbf{x} = N$ and $\mathbf{M}\mathbf{x} = \mathbf{0}$ ensure that the sought after OAs have N rows and all of their $J_r(\ell) = 0$ for all $\ell \subseteq \{1, \ldots, k\}$ such that $1 \leq |\ell| = r \leq t$. We added the redundant equalities $-\mathbf{M}\mathbf{x} = \mathbf{0}$ so that we can prove the following lemma.

Lemma 9. *Let $1 \leq t \leq k-1$ and $\mathbf{x}$ be the frequency vector of an OA$(N, k, 2, t)$ as in ILP (27). If t is even (odd), then the formulation symmetry group of ILP (27) contains $G(k)^{\mathrm{OD}}$ ($G^{\mathrm{iso}}(k, 2)$),* *where $g(\mathbf{x})$ is the frequency vector of an OD-equivalent (isomorphic) OA$(N,k,2,t)$ for each element g of $G(k)^{\mathrm{OD}}$ ($G^{\mathrm{iso}}(k,2)$).*

Proof. The group $G(k)^{\mathrm{OD}}$ ($G^{\mathrm{iso}}(k,2)$) acts on $N \times k$ arrays with symbols from $\{-1,1\}$. This action induces an action on the frequency vectors of such arrays. Identify the action of $G(k)^{\mathrm{OD}}$ ($G^{\mathrm{iso}}(k,2)$) on $\mathbf{x}$ by $g(x_{p_1}) = x_{p_2}$ if and only if

$$g((z_{p_1 1},\dots,z_{p_1 k})) = (z_{p_2 1},\dots,z_{p_2 k}),$$

where $p_1,p_2 \in \{1,\dots,2^k\}$ and g is an OD-equivalence (isomorphism) operation on the columns of the row vector $(z_{p_1 1},\dots,z_{p_1 k})$. Hence, g sends $\mathbf{Z}$ to one of its OD-equivalent (isomorphic) copies. Here, $\mathbf{x}$ is indexed by the rows of $\mathbf{Z}$. Since g is an invertible map from $\{-1,1\}^k$ to $\{-1,1\}^k$ and

$$\text{the set of columns of } \mathbf{Z}^T = \{-1,1\}^k,$$

g is a permutation of the rows of $\mathbf{Z}$. Consequently, g is a permutation of columns of $\mathbf{M}$. Now, since g is an OD-equivalence (isomorphism) operation on the columns of $\mathbf{Z}$, g's action on the columns of $\mathbf{M}$ sends the row $\mathbf{z}^T_{i_1,\dots,i_r}$ to a row of the form $\pm\mathbf{z}^T_{i'_1,\dots,i'_{r'}}$ ($\pm\mathbf{z}^T_{i''_1,\dots,i''_r}$), where $\{i_1,\dots,i_r\},\{i'_1,\dots,i'_{r'}\} \subseteq \{1,\dots,k\}$ ($\{i_1,\dots,i_r\},\{i''_1,\dots,i''_r\} \subseteq \{1,\dots,k\}$), and

$$r' = \begin{cases} r \text{ or } r+1 & \text{if } r \text{ is odd,} \\ r \text{ or } r-1 & \text{otherwise.} \end{cases}$$

Hence, when t is even (odd), we get

$$\begin{bmatrix} \mathbf{M} \\ -\mathbf{M} \end{bmatrix} g(\mathbf{x}) = \mathbf{\Pi} \begin{bmatrix} \mathbf{M} \\ -\mathbf{M} \end{bmatrix} \mathbf{x} = \mathbf{\Pi 0} = \mathbf{0},$$

where $\mathbf{\Pi}$ is a $\sum_{i=1}^t 2\binom{k}{i} \times \sum_{i=1}^t 2\binom{k}{i}$ permutation matrix. This proves that for even (odd) t the formulation symmetry group of ILP (27) contains $G(k)^{\mathrm{OD}}$ ($G^{\mathrm{iso}}(k,2)$) as g maps both the objective function $\mathbf{1}^T\mathbf{x}$ and the constraint $\mathbf{1}^T\mathbf{x} = N$ to themselves and permutes the constraints $\mathbf{x} \geq \mathbf{0}$ among each other. Finally, by Lemma 5 applied to the frequency vector $\mathbf{x}$ of an OA$(N,k,2,t)$ for $t \in \{1,\dots,k\}$ (as the action of each element of $G^{\mathrm{iso}}(k,2)$ sends the frequency vector of an OA$(N,k,2,t)$ to that of one of its isomorphic copies), $g(\mathbf{x})$ is the frequency vector of an OD-equivalent (isomorphic) OA$(N,k,2,t)$. □

Lemma 10. *For odd t and $1 \leq t \leq k-1$, $G(k,2,t)^{\mathrm{LP}}$ contains none of the R_i in Definition 8.*

Proof. As in the proof of Lemma 9, identify the action of $G(k)^{\mathrm{OD}}$ on the frequency vectors $\mathbf{x}$ of $N \times k$ arrays with symbols from $\{-1,1\}$. Pick $1 \leq i_1 < \dots < i_t \leq k$ and $i \in \{1,\dots,k\}\backslash\{i_1,\dots,i_t\}$. Take g in Lemma 9 to be R_i and observe that R_i is a linear transformation from $\mathbb{R}^{2^k}$ to $\mathbb{R}^{2^k}$ as R_i is a permutation of the coordinates of $\mathbf{x}$. Let $\mathbf{R}_i$ be the matrix of R_i with respect to the standard basis. Then $\mathbf{M}\mathbf{R}_i$ has the $(t+1)$-way Hadamard product $(\mathbf{z}_i \odot (\mathbf{z}_{i_1} \odot \dots \odot \mathbf{z}_{i_t}))^T$ as one of its rows, and this row is orthogonal to all the rows of $\mathbf{M}$. Hence, the action of R_i on the constraints of ILP (27) produces an equality constraint that is not a linear combination of the original constraints, so $R_i \notin G(k,2,t)^{\mathrm{LP}}$. □

The following theorem is in part stated, but not proven in [1].

Lemma 11. *Let $1 \leq t \leq k-1$. Then $G(k,2,t)^{\mathrm{LP}}$ contains $G(k)^{\mathrm{OD}} \cong S_2^k \rtimes S_{k+1}$, where each element of $G(k)^{\mathrm{OD}}$ sends the frequency vector of an OA$(N,k,2,t)$ to that of an OD-equivalent OA$(N,k,2,t)$ if and only if t is even. Hence, for even t, $|G(k,2,t)^{\mathrm{LP}}| \geq |S_2^k \rtimes S_{k+1}| = (k+1)!2^k$.*

Proof. The equality constraints of ILP (27) can be obtained as linear combinations of those of ILP (24). Rows of $\mathbf{M}$ and $\mathbf{1}^T$ form a mutually orthogonal set of vectors of size $\sum_{i=0}^{t}\binom{k}{i}$ in $\mathbb{R}^{2^k}$. Hence, the rank of the equality constraint matrix of ILP (27) is $\sum_{i=0}^{t}\binom{k}{i}$. By Lemma 1 in [32], $\sum_{i=0}^{t}\binom{k}{i}$ is also the rank of the equality constraint matrix of ILP (24). Now, this implies that the equality constraints of ILPs (24) and (27) are row equivalent. Both ILPs (24) and (27) have the same set of non-negative variables without having any other inequality constraints. Hence, the LP relaxations of both ILPs have the same feasible set and objective function, and consequently the same symmetry group $G(k,2,t)^{\mathrm{LP}}$. Now, if t is even, then by Lemma 9 the formulation symmetry group of ILP (27) contains $G(k)^{\mathrm{OD}}$. Since the formulation symmetry group of ILP (27) is a subgroup of $G(k,2,t)^{\mathrm{LP}}$, $G(k,2,t)^{\mathrm{LP}}$ contains $G(k)^{\mathrm{OD}}$. The converse statement follows from Lemma 10. □

Now, inequality (26) follows from Lemmas 8 and 11. Theorem 8 follows from Remark 2, Lemmas 8 and 11, and comparing group sizes. By taking $\mathbf{x} = (N/s^k)\mathbf{1}$, we see that the LP relaxation of ILP (24) is feasible, so Method 5 of Section 3 applies.

The following lemma identifies an LP in standard form that has the same feasible set as the LP relaxation of ILP (24).

Lemma 12. *Let $1 \le t \le k-1$, $p_{\max} \neq \lambda/(s^{k-t})$, and $\mathbf{A}'(k,s,t)\mathbf{x} = \mathbf{b}'(k,s,t)$ be the equality constraints of ILP (25), where $p_{\max} \le \lambda$ is the upper bound for the variables in ILP (24). Let*

$$\left[\mathbf{B}'(k,s,t) \mid \mathbf{d}'(k,s,t)\right] = \begin{cases} \left[-\mathbf{I} \mid \mathbf{0}\right] & \text{if } x_1 \ge 0 \text{ is a facet and } x_1 \le p_{\max} \text{ is not a facet of the LP relaxation of ILP (24)}, \\ \left[\mathbf{I} \mid p_{\max}\mathbf{1}\right] & \text{if } x_1 \le p_{\max} \text{ is a facet and } x_1 \ge 0 \text{ is not a facet of the LP relaxation of ILP (24)}, \\ \left[\begin{array}{r|r} -\mathbf{I} & \mathbf{0} \\ \mathbf{I} & p_{\max}\mathbf{1} \end{array}\right] & \text{otherwise.} \end{cases}$$

Then

$$\begin{aligned} \min\ & \mathbf{1}^T\mathbf{x} \\ s.t.\ & \mathbf{A}'(k,s,t)\mathbf{x} = \mathbf{b}'(k,s,t), \\ & \mathbf{B}'(k,s,t)\mathbf{x} \le \mathbf{d}'(k,s,t) \end{aligned} \tag{28}$$

is an LP in standard form with the same feasible set and objective function as the LP relaxation of ILP (24).

Proof. Let $\mathbf{A}(k,s,t)\mathbf{x} = \mathbf{b}(k,s,t)$ be the equality constraints of ILP (24). It is easy to see that the equality constraints $\mathbf{A}(k,s,t)\mathbf{x} = \mathbf{b}(k,s,t)$ can be obtained as linear combinations of the equality constraints $\mathbf{A}'(k,s,t)\mathbf{x} = \mathbf{b}'(k,s,t)$ and vice versa. By Lemma 2 in [32],

$$\operatorname{rank}(\mathbf{A}(k,s,t)) = \sum_{i=0}^{t}\binom{k}{i} = m,$$

and m is also equal to the number of rows of $\mathbf{A}'(k,s,t)$. Hence $\mathbf{A}'(k,s,t)$ has full row rank. Then it suffices to show that either each inequality in $\mathbf{0} \le \mathbf{x}$, and/or each inequality in $\mathbf{x} \le p_{\max}\mathbf{1}$ is a facet of the LP relaxation of ILP (24).

Let LP (24) be the LP relaxation of ILP (24) and $\mathcal{F}$ be its feasible set. Since $G(k,s,t)$ preserves $\mathcal{F}$, it sends facets of LP (24) to its facets and preserves the set of all equality constraints satisfied by each point in $\mathcal{F}$. Moreover, $G(k,s,t)$ acts transitively on the variables x_i. Thus, it acts transitively on the inequalities $0 \le x_i$ as well as on $x_i \le p_{\max}$. So, if one of the inequalities is satisfied as an equality by each point in $\mathcal{F}$, then either $x_i = 0$ for $i \in \{1,\ldots,s^k\}$ or $x_i = p_{\max}$ for $i \in \{1,\ldots,s^k\}$

for each $\mathbf{x} \in \mathcal{F}$. We cannot have $x_i = 0$ for $i \in \{1, \ldots, s^k\}$ or $x_i = p_{\max}$ for $i \in \{1, \ldots, s^k\}$ when $p_{\max} \neq \lambda/(s^{k-t})$ as such points are not in $\mathcal{F}$. Therefore, no inequality of LP (24) can be satisfied as an equality by each point in $\mathcal{F}$. Moreover, at least one of the inequalities of LP (24) is a facet. Otherwise the feasible set would not be bounded. Then, since $G(k,s,t)$ acts transitively on the variables x_i and preserves the set of facets of LP (24), $0 \leq x_i$ for $i \in \{1, \ldots, s^k\}$ and/or $x_i \leq p_{\max}$ for $i \in \{1, \ldots, s^k\}$ are all facets of LP (24). □

Let S_{s^k} be the group of all permutations of coordinates of vectors in $\mathbb{R}^{s^k}$. By Lemma 12 and the fact that $(N/s^k)\mathbf{1} \in \mathcal{T}^{\mathrm{LP}(28)}_{\mathrm{Fix}_H}$ for any subgroup H of S_{s^k}, we get

$$G(k,s,t)^{\mathrm{LP}} = G^{\mathrm{Null}}_{(\mathbf{A}'(k,s,t),\mathbf{B}'(k,s,t),\mathbf{d}'(k,s,t),\mathbf{1})} = G_{\mathbf{P}_{\mathbf{A}'(k,s,t)^T}},$$

and Theorem 7 follows.

Method 3 can be used to compute the automorphism group $G_{\mathbf{P}_{\mathbf{A}'(k,s,t)^T}}$ of $\mathbf{P}_{\mathbf{A}'(k,s,t)^T}$ by taking $\mathbf{A} := \mathbf{A}'(k,s,t)$, $\mathbf{c} := \mathbf{1}$ as inputs and stopping once the Step 24 computation finishes. Then, the output is $H_{\mathbf{P}_{\mathbf{A}^T}} = G_{\mathbf{P}_{\mathbf{A}^T}} = G_{\mathbf{P}_{\mathbf{A}'(k,s,t)^T}}$. By using Method 3, we computed $G(k,s,t)^{\mathrm{LP}} = G_{\mathbf{P}_{\mathbf{A}'(k,s,t)^T}}$ for many k,s,t combinations. Our computational results and Theorems 9 and 10 suggest that inequality (26) and the containments in Lemmas 8 and 11 are in fact equalities for $1 \leq t \leq k-1$ unless $s = 2$ and $k = t+1$. (In [1], it was proven that $G(3,2,2)^{\mathrm{LP}} \cong (S_4 \times S_4) \rtimes S_2$ by using `GAP` and Method 3 of Section 3, where $|G(3,2,2)^{\mathrm{LP}}| = 1,152 > 2^3 4! = 192$. By using Method 3 of Section 3, we also observed that $|G(t+1,2,t)^{\mathrm{LP}}| > 2^{t+1}(t+2)!$ for $t = 3, \ldots, 10$ and $(|G(t+1,2,t)^{\mathrm{LP}}|)/(2^{t+1}(t+2)!)$ increases exponentially with t.) In fact, the only known cases for which $1 \leq t \leq k-1$ and yet inequality (26) is not satisfied as an equality are when $k = t+1$ and $s = 2$.

We excluded the case $t = k$ from our results. This is because this case is trivial and completely solved by the following remark.

Remark 3. *When $t = k$, $x_i = \lambda$ for $i \in \{1, \ldots, s^k\}$ is the unique solution to ILP (24). Consequently, the symmetry group of ILP (24) in this case is S_{s^k}, where S_{s^k} is the set of all permutations of the variables (frequencies) in ILP (24).*

6. Computational experiments

A speed comparison of exploiting $G(k,s,t)$ and $G(k,s,t)^{\mathrm{LP}} = G_{\mathbf{P}_{\mathbf{A}'(k,s,t)^T}}$ for ILP (25) and $G(k,s,t)^{\mathrm{LP}\leq}$ for the Theorem 1 ILP in [10] within B&B with isomorphism pruning [19] is made in Table 1. The groups $G(k,s,t)$ and $G(k,s,t)^{\mathrm{LP}\leq}$ were computed by using the method in [20, 30] as formulation symmetry groups of ILP (24) and Theorem 1 ILP in [10]. The group $G(k,s,t)^{\mathrm{LP}} = G_{\mathbf{P}_{\mathbf{A}'(k,s,t)^T}} = H_{\mathbf{P}_{\mathbf{A}'(k,s,t)^T}}$ was computed by using Method 3 as described at the end of Section 5. The automorphism group in each of these methods was computed by using `nauty 25.1` [25, 26]. A computer program written in `C` was used for computing *nce* in Step 4 and constructing the edge colored graph between Step 6 and Step 14 in Method 3. A singular value decomposition $\mathbf{UDV}^T$ and $\mathbf{P}_{\mathbf{A}'(k,s,t)^T} = \mathbf{VI}^{(p)}_n\mathbf{V}^T$ in Step 2 were computed in `MATLAB 8.0` [21]. `ISOP 1.1` implementation [19] that calls the `CPLEX 12.5.1` libraries [7] was used for B&B with isomorphism pruning. The overall running times and the numbers of non-isomorphic solutions pertaining to exploiting $G(k,s,t)$ and $G(k,s,t)^{\mathrm{LP}\leq}$ except the OA(160, 8, 2, 4) and OA(176, 8, 2, 4) cases (second, fourth, fifth, and seventh columns) are copied from [10]. All cases were run on an `HP Z820` workstation with 64GB of RAM and a 3.10 GHz `Intel(R) Xeon(R) E5-2687W` processor. (Processor information in exploiting $G(k,s,t)$ and $G(k,s,t)^{\mathrm{LP}\leq}$ for the results in [10] that we provide here was not provided in [10].) For each OA(N,k,s,t), the second and the third columns report the number of non-isomorphic solutions enumerated for ILP (25) using $G(k,s,t)$ and $G(k,s,t)^{\mathrm{LP}}$. (These are also the number of non-isomorphic solutions of ILP (24) using $G(k,s,t)$ and $G(k,s,t)^{\text{LP}}$.) The fourth column reports the number of non-isomorphic solutions found from the Theorem 1 ILP formulation in [10] using $G(k,s,t)^{\text{LP}\leq}$. The fifth, sixth, and the seventh columns report the times it took to find all non-isomorphic solutions using $G(k,s,t)$, $G(k,s,t)^{\text{LP}}$, and $G(k,s,t)^{\text{LP}\leq}$ with ILP (25), ILP (25), and the Theorem 1 ILP formulation in [10]. Each of these times includes the time it took to compute the exploited symmetry group. The times in parentheses, on the other hand, are the times needed to compute the corresponding symmetry groups. For most N,k,s,t cases in Table 1, the time needed to find all non-isomorphic solutions is much greater than that for computing the corresponding symmetry groups.

Table 1: Speed comparisons and the number of non-isomorphic solutions

$\text{OA}(N,k,s,t)$	ILP (25) $G(k,s,t)$ # of OAs	ILP (25) $G(k,s,t)^{\text{LP}}$ # of OAs	ILP in [10] $G(k,s,t)^{\text{LP}\leq}$ # of OAs	ILP (25) $G(k,s,t)$ Times (sec.)	ILP (25) $G(k,s,t)^{\text{LP}}$ Times (sec.)	ILP in [10] $G(k,s,t)^{\text{LP}\leq}$ Times (sec.)
OA(20,6,2,2)	75	23	3,069	1 (0)	7 (6)	64 (6)
OA(20,7,2,2)	474	102	51,695	13 (0)	9 (6)	2,578 (7)
OA(20,8,2,2)	1,603	211	383,729	109 (1)	22 (7)	66,377 (11)
OA(20,9,2,2)	2,477	351	1,157,955	485 (4)	67 (14)	879,382 (26)
OA(20,10,2,2)	2,389	260	$\geq 28,195$	1,684 (33)	215 (72)	$\geq 37,214$ (76)
OA(24,5,2,2)	63	31	723	1 (0)	10 (6)	18 (6)
OA(24,6,2,2)	1,350	274	62,043	22 (0)	12 (6)	1,381 (6)
OA(24,7,2,2)	57,389	7,990	6,894,001	1,721 (0)	257 (6)	428,220 (7)
OA(24,8,2,2)	1,470,157	165,596	4,505,018	99,738 (1)	10,082 (7)	653,671 (11)
OA(24,9,2,2)	3,815,882	1,309,475	-	763,643 (4)	223,138 (14)	- (25)
OA(24,5,2,3)	1	1	2	0 (0)	6 (6)	12 (6)
OA(24,6,2,3)	2	2	5	0 (0)	7 (6)	12 (6)
OA(24,7,2,3)	1	1	5	0 (0)	9 (6)	16 (8)
OA(24,8,2,3)	1	1	6	1 (1)	14 (7)	23 (13)
OA(24,9,2,3)	1	1	6	6 (4)	26 (13)	44 (30)
OA(24,10,2,3)	1	1	5	55 (42)	104 (57)	129 (91)
OA(24,11,2,3)	1	1	3	520 (359)	540 (441)	461 (320)
OA(32,6,2,3)	10	10	31	2 (0)	8 (6)	12 (6)
OA(32,7,2,3)	17	17	76	2 (0)	8 (6)	16 (8)
OA(32,8,2,3)	33	33	194	7 (1)	14 (7)	77 (13)
OA(32,9,2,3)	34	34	364	24 (5)	33 (13)	658 (30)
OA(32,10,2,3)	32	32	561	102 (42)	112 (56)	7,338 (91)
OA(32,11,2,3)	22	22	≥ 441	560 (364)	597 (442)	$\geq 36,463$ (319)
OA(40,6,2,3)	9	9	65	1 (0)	7 (6)	13 (6)
OA(40,7,2,3)	25	25	580	2 (0)	9 (6)	41 (8)
OA(40,8,2,3)	105	105	6,943	20 (1)	27 (7)	4,178 (13)
OA(40,9,2,3)	213	213	43,713	206 (5)	215 (13)	260,919 (30)
OA(40,10,2,3)	353	353	$\geq 1,511$	1,765 (42)	1,694 (57)	$\geq 36,279$ (91)
OA(48,7,2,3)	397	397	13,469	34 (0)	40 (7)	862 (8)
OA(48,8,2,3)	8,383	8,383	896,963	2,232 (1)	2,237 (8)	552,154 (13)
OA(54,5,3,3)	4	4	49	2 (1)	10 (7)	36 (13)
OA(54,6,3,3)	0	0	0	17 (13)	37 (24)	167 (53)
OA(56,6,2,3)	86	86	1,393	4 (0)	11 (6)	36 (6)
OA(56,7,2,3)	4,049	4,049	285,184	443 (0)	450 (6)	20,415 (8)
OA(64,7,2,4)	7	4	21	99 (0)	260 (6)	15 (8)
OA(64,8,2,4)	3	2	10	12 (1)	38 (8)	23 (14)
OA(80,6,2,4)	1	1	6	1 (0)	7 (6)	12 (7)
OA(80,7,2,4)	0	0	0	0 (0)	8 (7)	15 (8)
OA(81,5,3,4)	1	1	2	16 (1)	23 (7)	20 (13)
OA(96,7,2,4)	4	2	31	3 (0)	10 (6)	15 (8)
OA(96,8,2,4)	0	0	0	2 (1)	11 (8)	60 (15)
OA(112,6,2,4)	3	2	25	1 (0)	8 (6)	13 (6)
OA(112,7,2,4)	0	0	0	1 (0)	8 (6)	18 (8)
OA(144,8,2,4)	20	7	3,392	1,793 (1)	774 (8)	1,535,314 (14)
OA(160,8,2,4)	99,618	11,712	-	123,180 (1)	32,880 (9)	- (14)
OA(176,8,2,4)	1,157,443	129,138	-	- (1)	1,067,822 (8)	- (14)
OA(162,6,3,4)	0	0	0	20 (14)	32 (24)	267 (62)

The set of all non-isomorphic solutions under the action of $G(k,s,t)$ and $G(k,s,t)^{\text{LP}}$ correspond to a set of all non-isomorphic and non-OD-equivalent $\text{OA}(N,k,s,t)$. The numbers of all non-isomorphic solutions obtained by exploiting $G(k,s,t)$ and $G(k,s,t)^{\text{LP}}$ for the bottleneck cases $\text{OA}(160,8,2,4)$ and $\text{OA}(176,8,2,4)$ corroborate the numbers of all non-isomorphic and non-OD-equivalent $\text{OA}(160,8,2,4)$ and $\text{OA}(176,8,2,4)$ in [6]. The number of all non-isomorphic solutions under the action of $G(k,s,t)^{\text{LP}\leq} \cong S_{s-1} \wr S_k$ equals the number of all $\text{OA}(N,k,s,t)$ up to a weaker form of isomorphism.

For cases in which $G(k,s,t)^{\text{LP}}$ captures symmetries not in $G(k,s,t)$, the speedup gleaned from adding slack variables to the Theorem 1 ILP formulation in [10] and using $G(k,s,t)^{\text{LP}}$ to enumerate $\text{OA}(N,k,s,t)$ up to OD-equivalence under the action of $G(k,s,t)^{\text{LP}}$ with ILP (25) becomes the fastest enumeration method for OAs as the number of variables s^k increases. (ILP (25) can be obtained from the Theorem 1 ILP formulation in [10] by adding slack variables and the $x_1 \geq 1$ inequality.) Moreover, this speedup appears to grow exponentially with the number of variables. However, the cases OA$(64, 7, 2, 4)$ and OA$(24, 11, 2, 3)$ are exceptions to this trend. Hence, exploiting a larger symmetry group drastically overcomes the extra computational burden due to having additional variables. This underscores the importance of developing tools for finding larger subgroups of the symmetry group of an ILP. Finally, the cost of computing $G(k, 2, 2)^{\text{LP}}$ when $k \geq 6$ and $G(k, 2, 4)^{\text{LP}}$ when $k \geq 8$ is more than compensated for with the speedup gleaned from exploiting the additional symmetries not in $G(k, 2, 2)$ and $G(k, 2, 4)$. For many $s = 2$ and even t cases and all the bottleneck cases, using the larger symmetry group $G(k, 2, t)^{\text{LP}}$ drastically reduces solution times.

A set of all non-isomorphic (non-OD-equivalent if $s = 2$ and t is even) OA(N, k, s, t) can be obtained by adding columns to a set of all non-isomorphic (non-OD-equivalent if $s = 2$ and t is even) OA$(N, k-1, s, t)$. Bulutoglu and Ryan [6] used this fact to develop the Hybrid method that enumerates a set of all non-isomorphic (non-OD-equivalent) OA(N, k, s, t) by adding columns to a set of all non-isomorphic (non-OD-equivalent) OA$(N, k-1, s, t)$. This method adds columns to input OA$(N, k-1, s, t)$ by finding a set of all non-isomorphic solutions to ILPs derived from the input OA$(N, k-1, s, t)$ and ILP (25). For each input OA$(N, k-1, s, t)$, it uses B&B with isomorphism pruning with a group depending on the input OA$(N, k-1, s, t)$. However, this method removes only some of the symmetry within the B&B with isomorphism pruning algorithm and requires converting OA(N, k, s, t) to graphs and using `nauty` [25, 26] for removing isomorphic graphs that correspond to isomorphic (OD-equivalent) OA(N, k, s, t) [6, 34].

McKay [24, 27] had previously developed a technique for generating combinatorial objects with partial isomorph rejection when it is possible to sequentially obtain larger objects from the smaller. In particular, McKay's technique is applicable to generating a set of all non-isomorphic (non-OD-equivalent) OA(N, k, s, t) from a set of all non-isomorphic (non-OD-equivalent) OA$(N, k-1, s, t)$. However, just like the Hybrid method, when this technique is applied to the problem of generating a set of all non-isomorphic (non-OD-equivalent) OA(N, k, s, t) it does not completely eliminate the need to use `nauty` for removing isomorphic (OD-equivalent) OA(N, k, s, t). In fact, Bulutoglu and Ryan [6] implemented McKay's technique for generating a set of all non-OD-equivalent OA$(160, 8, 2, 4)$ and OA$(176, 8, 2, 4)$ from a set of all non-OD-equivalent OA$(160, 7, 2, 4)$ and OA$(176, 7, 2, 4)$ and observed that the running times for the Hybrid method were $1/(11.44)$ and $1/(1.44)$ times those of McKay's technique. The OA$(160, 8, 2, 4)$ and OA$(176, 8, 2, 4)$ are the largest 2-symbol, strength 4 OAs that have been classified, where the use of a symmetry exploiting method was necessary [6].

Unlike the Hybrid method or McKay's technique, exploiting $G(k, 2, t)^{\text{LP}}$ when t is even within B&B with isomorphism pruning enabled us to directly generate a set of all non-OD-equivalent OA$(N, k, 2, t)$ without using `nauty` [25, 26] to remove OD-equivalent OAs. However, we did use `nauty` [25, 26] to find $G(k, s, t)^{\text{LP}}$. This was a viable method by Theorem 8, and it reduced the enumeration times of a set of all non-OD-equivalent OA$(160, 8, 2, 4)$ and OA$(176, 8, 2, 4)$ in comparison to the Hybrid method in [6] by factors of $1/(2.16)$ and $1/(1.36)$. (We ran the OA$(176, 8, 2, 4)$ case on our `HP Z820` workstation with 128GB of RAM and 2.00 GHz `Intel(R) Xeon(R) E5-2650` processor as well to allow making comparisons to the corresponding times for the Hybrid method and McKay's technique in [6].) Hence, using $G(k, 2, t)^{\text{LP}}$ as described in this paper reduced the running time for finding all OD-equivalence classes of OA$(160, 8, 2, 4)$ and OA$(176, 8, 2, 4)$ by factors of $1/(24.71)$ and $1/(1.96)$ in comparison to McKay's technique.

7. Conclusion

In this paper, we showed that there may be hidden symmetries in an LP that cannot be captured by the formulation symmetry group. These symmetries are either masked by redundant constraints or due to equality constraints. As a remedy, we developed a method that captures all the symmetries of a feasible LP. (The symmetry group of an infeasible LP is isomorphic to

S_n.) We tested our method on the LP relaxations of a family of ILPs for classifying OAs, and for OA$(N, k, 2, t)$ with even t, we found LP relaxation symmetry groups with drastically larger sizes than their corresponding formulation symmetry groups. Finally, we exploited the newly found larger groups $G(k, 2, t)^{\mathrm{LP}}$ within B&B with isomorphism pruning. This enabled us to improve the times it took to find all OD-equivalence classes of OA$(160, 8, 2, 4)$ and OA$(176, 8, 2, 4)$ by factors of $1/(2.16)$ and $1/(1.36)$.

One of the key findings of this article involves the enumeration of a set of all non-isomorphic solutions to an ILP. In this context, converting the inequality constraints to equalities by introducing slack variables and using the LP relaxation G^{LP} of the resulting ILP within B&B with isomorphism pruning can reduce the enumeration time by several orders of magnitude. In particular, this method would be most useful in determining whether a given ILP is feasible. We propose testing this idea along with the methods in this paper on the MIPLIP problems studied in [13, 30] as a future research project. A limited preliminary study on the MIPLIB problems in [13] suggests that, when computing G^{LP}, generating the graphs between Step 6 and Step 14 in Method 3 and computing the double coset decompositions in Method 4 are the major bottlenecks in terms of both time and memory requirements. Based on our experience with the OA problem, we expect that the time requirements will be much greater for finding sets of all non-isomorphic optimal solutions of the MIPLIB problems under the action of their respective LP relaxation symmetry groups than that for computing their LP relaxation symmetry groups.

Acknowledgements

The authors thank two anonymous referees for improving the paper. The authors also thank Mr. David Doak for general computer support. This research was supported by the AFOSR grant F4FGA04013J001. The views expressed in this article are those of the authors and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government.

References

[1] Arquette, D. M., Bulutoglu, D. A., 2016. The linear programming relaxation permutation symmetry group of an orthogonal array defining integer linear program. LMS Journal of Computation and Mathematics 19 (1), 206–216.

[2] Banciu, M., 2011. Dual simplex. In: Cochran, J. J., Cox, L. A., Keskinocak, P., Kharoufeh, J. P., Smith, C. J. (Eds.), Wiley Encyclopedia of Operations Research and Management Science. John Wiley & Sons, Inc., Hoboken, NJ, USA.

[3] Bödi, R., Herr, K., Joswig, M., 2013. Algorithms for highly symmetric linear and integer programs. Mathematical Programming Series A 137 (1-2), 65–90.

[4] Bremner, D., Sikirić, M. D., Pasechnik, D. V., Rehn, T., Schürmann, A., 2014. Computing symmetry groups of polyhedra. LMS Journal of Computation and Mathematics 17 (1), 565–581.

[5] Bulutoglu, D. A., Margot, F., 2008. Classification of orthogonal arrays by integer programming. Journal of Statistical Planning and Inference 138 (3), 654–666.

[6] Bulutoglu, D. A., Ryan, K. J., 2018. Integer programming for classifying orthogonal arrays. Australasian Journal of Combinatorics 70 (3), 362–385.

[7] CPLEX, 2013. Cplex, version 12.5.1.
URL `https://www.cplex.com`

[8] Egan, J., Wanless, I. M., 2016. Enumeration of MOLS of small order. Mathematics of Computation 85, 799–824.

[9] GAP, 2013. GAP – Groups, Algorithms, and Programming, Version 4.6.4. The GAP Group.
URL `https://www.gap-system.org`

[10] Geyer, A. J., Bulutoglu, D. A., Rosenberg, S. J., 2014. The LP relaxation orthogonal array polytope and its permutation symmetries. Journal of Combinatorial Mathematics and Combinatorial Computing 91, 165–176.

[11] Hoffman, C. M., 1982. Group-Theoretic Problems and Graph Isomorphism. Lecture Notes in Computer Science 136. Springer Verlag, New York, NY, USA.
[12] Leon, S. J., 2002. Linear Algebra with Applications, 6th Edition. Prentice Hall, Upper Saddle River, NJ, USA.
[13] Liberti, L., 2012. Reformulations in mathematical programming: Automatic symmetry detection and exploitation. Mathematical Programming Series A 131 (1-2), 273–304.
URL http://www.lix.polytechnique.fr/~liberti/minlpgroup.pdf
[14] Loos, R., et al., 2003. Topics of computer algebra. In: Grabmeier, J., Kaltofen, E., Weispfenning, V. (Eds.), Computer Algebra Handbook Foundations Applications Systems. Springer-Verlag, Berlin, Heidelberg, Germany, pp. 11–162.
[15] Luks, E. M., 1993. Permutation groups and polynomial-time computation. In: Finkelstein, L., Kantor, W. (Eds.), DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Groups and Computation. Vol. 11. American Mathematical Society, pp. 139–175.
[16] Margot, F., 2002. Pruning by isomorphism in branch-and-cut. Mathematical Programming Series A 94 (1), 71–90.
[17] Margot, F., 2003. Exploiting orbits in symmetric ILP. Mathematical Programming Series B 98 (1-3), 3–21.
[18] Margot, F., 2003. Small covering designs by branch-and-cut. Mathematical Programming Series B 94 (2-3), 207–220.
[19] Margot, F., 2007. Symmetric ILP: Coloring and small integers. Discrete Optimization 4 (1), 40–62.
[20] Margot, F., 2010. Symmetry in integer linear programming. In: Junger, M., Liebling, T., Naddef, D., Nemhauser, G. L., Pulleybank, W., Reinelt, G., Rinaldi, G., Wolsey, L. (Eds.), 50 Years of Integer Programming 1958-2008. Springer-Verlag, Berlin, Heidelberg, Germany, pp. 647–686.
[21] MATLAB, 2012. version 8.0 (R2012b). The MathWorks Inc., Natick, Massachusetts.
[22] McKay, B. D., 1979. Hadamard equivalence via graph isomorphism. Discrete Mathematics 27 (2), 213–214.
[23] McKay, B. D., 1981. Practical graph isomorphism. Congressus Numerantium 30, 45–87.
[24] McKay, B. D., 1998. Isomorph-free exhaustive generation. Journal of Algorithms 26 (2), 306–324.
[25] McKay, B. D., Piperno, A., 2013. Nauty User's Guide (version 2.5).
URL http://users.cecs.anu.edu.au/~bdm/nauty/
[26] McKay, B. D., Piperno, A., 2013. Practical graph isomorphism, II. Journal of Symbolic Computation 60, 94–112.
[27] McKay, B. D., Radziszowski, S. P., 1996. The nonexistence of 4-$(12, 6, 6)$ designs. In: Wallis, W. D. (Ed.), Computational and Constructive Design Theory. Kluwer Academic Publishers, Dordrecht, Netherlands, pp. 177–188.
[28] Nash, S. G., Sofer, A., 1996. Linear and Nonlinear Programming. McGraw Hill, New York, NY, USA.
[29] Ostrowski, J., Linderoth, J., Rossi, F., Smriglio, S., 2008. Constraint orbital branching. In: Lodi, A., Panconesi, A., Rinaldi, G. (Eds.), Integer Programming and Combinatorial Optimization. IPCO 2008. Lecture Notes in Computer Science. Vol. 5035. Springer-Verlag, Berlin, Heidelberg, Germany, pp. 225–239.
[30] Pfetsch, M., Rehn, T., 2018. Computational comparison of symmetry handling methods for mixed integer programs. Mathematical Programming Computation.
URL https://doi.org/10.1007/s12532-018-0140-y
[31] Rehn, T., 2010. Fundamental permutation group algorithms for symmetry computation. Diploma thesis, Otto-von-Guericke University Magdeburg.
[32] Rosenberg, S. J., 1995. A large index theorem for orthogonal arrays, with bounds. Discrete Mathematics 137 (1), 315–318.
[33] Rotman, J. J., 1994. An Introduction to the Theory of Groups, 4th Edition. Springer-Verlag, New York, NY, USA.
[34] Ryan, K. J., Bulutoglu, D. A., 2010. Minimum aberration fractional factorial designs with large N. Technometrics 52 (2), 250–255.
[35] Stufken, J., Tang, B., 2007. Complete enumeration of two-level orthogonal arrays of strength d with $d + 2$ constraints. Annals of Statistics 35 (2), 793–814.
[36] Wolsey, L. A., 1998. Integer Programming, 1st Edition. Wiley, New York, NY, USA.