



HAL
open science

Sub-Symmetry-Breaking Inequalities for ILP with Structured Symmetry

Pascale Bendotti, Pierre Fouilhoux, Cécile Rottner

► **To cite this version:**

Pascale Bendotti, Pierre Fouilhoux, Cécile Rottner. Sub-Symmetry-Breaking Inequalities for ILP with Structured Symmetry. IPCO 2019 - 20th Conference on Integer Programming and Combinatorial Optimization, May 2019, Ann Arbor, Michigan, United States. pp.57-71, 10.1007/978-3-030-17953-3_5. hal-01876358

HAL Id: hal-01876358

<https://hal.science/hal-01876358v1>

Submitted on 18 Sep 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Sub-symmetry-breaking inequalities and application to the Unit Commitment Problem

September 18, 2018

Abstract

We consider integer linear programs whose solutions are binary matrices and whose (sub-)symmetry groups are symmetric groups acting on (sub-)columns. We propose a framework to build (sub-)symmetry breaking inequalities for such problems, by introducing one additional variable per sub-symmetry group considered. The proposed framework is applied to derive such inequalities when the symmetry group is the symmetric group acting on the columns. It is also applied to derive inequalities breaking both symmetries and sub-symmetries in the Min-up/min-down Unit Commitment Problem (MUCP). We show the effectiveness of the approach by presenting an experimental comparison with state-of-the-art symmetry-breaking formulations for the MUCP with or without ramp constraints.

1 Introduction

We consider an Integer Linear Program (ILP) of the form

$$(ILP) \quad \min \left\{ c(x) \mid x \in \mathcal{X} \right\}, \text{ with } \mathcal{X} \subseteq \mathcal{P}(m, n) \text{ and } c : \mathcal{P}(m, n) \rightarrow \mathbb{R}$$

where $\mathcal{P}(m, n)$ is the set of $m \times n$ binary matrices. A symmetry is defined as a permutation π of the indices $\{(i, j) \mid 1 \leq i \leq m, 1 \leq j \leq n\}$ such that for any solution matrix $x \in \mathcal{X}$, matrix $\pi(x)$ is also solution and has same cost, *i.e.*, $\pi(x) \in \mathcal{X}$ and $c(x) = c(\pi(x))$. The *symmetry group* \mathcal{G} of (ILP) is the set of all such permutations. It partitions the solution set \mathcal{X} into *orbits*, *i.e.*, two matrices are in the same orbit if there exists a permutation in \mathcal{G} sending one to the other. In [4], symmetries arising in solution subsets of (ILP) are called *sub-symmetries*.

In this article, we focus on (sub-)symmetry groups containing all sub-column permutations of a given solution submatrix. These symmetry groups are assumed to be known or previously detected [16, 5].

Various techniques, so called *symmetry-breaking techniques*, are available to handle symmetries in (ILP) . The general idea is, in each orbit, to pick one solution, defined as the *representative*, and then restrict the solution set to the set of all representatives. The most common choice of representative is based on the lexicographical order. Column $y \in \{0, 1\}^m$ is said to be *lexicographically greater than* column $z \in \{0, 1\}^m$ if there exists $i \in \{1, \dots, m-1\}$ such that $\forall i' \leq i, y_{i'} = z_{i'}$ and $y_{i+1} > z_{i+1}$, *i.e.*, $y_{i+1} = 1$ and $z_{i+1} = 0$. We write $y \succ z$ (resp. $y \succeq z$) if y is lexicographically greater than z (resp. greater than or equal to z). A technique is said to be *full symmetry-breaking*

(resp. *partial symmetry-breaking*) if the solution set is exactly (resp. partially) restricted to the representative set.

Symmetry-breaking techniques can be based on specific branching and pruning rules during the B&B search [20, 23, 4]. Alternatively, variable aggregation can be a very efficient technique [15]. In general, aggregating symmetric variables in (*ILP*) yields a relaxation of (*ILP*) [7]. When the integer decomposition property [1] holds, *i.e.*, the aggregated solution can be disaggregated, aggregated (*ILP*) has same optimal value as (*ILP*).

Other symmetry-breaking techniques rely on full or partial symmetry-breaking inequalities. Techniques based on symmetry-breaking inequalities are *flexible*, *i.e.*, at any node of the B&B tree, the branching rule can be derived from any linear inequality on the variables. Most techniques based on branching and pruning rules are either full symmetry-breaking or flexible. Variable fixing [12, 4] is both full symmetry-breaking and flexible. The size of the LP solved at each node of the branching tree is generally invariant with pruning and variable fixing techniques. This size is decreased (resp. increased) by the use of variable aggregation (resp. symmetry-breaking inequalities) techniques.

Symmetry-breaking inequalities can be derived from the linear description of the convex hull of an arbitrary representative set [8]. In most works, each chosen representative x is lexicographically maximal in its orbit, *i.e.*, $x \succeq g(x)$, for each $g \in \mathcal{G}$. The convex hull of the representative set is called the symmetry-breaking polytope [9]. When x is a matrix and symmetry group \mathcal{G} acts on the columns of x , the symmetry-breaking polytope is called *orbitope*. Even if complete linear descriptions for symmetry-breaking polytopes may be hard to reach in general, integer programming formulations for these polytopes still yield full symmetry-breaking inequalities [9]. Instead of considering orbits of solutions, [16, 17] introduce inequalities enforcing a lexicographical order within orbits of variables.

If symmetry group \mathcal{G} is the *symmetric group* \mathfrak{S}_n acting on the columns, the chosen representative x of an orbit may be such that its columns $x(1), \dots, x(n)$ are lexicographically non-increasing, *i.e.*, for all $j < n$, $x(j) \succeq x(j+1)$. The convex hull of all $m \times n$ binary matrices with lexicographically non-increasing columns is called the *full orbitope* [13]. Full symmetry-breaking inequalities are introduced by Friedman [8]:

$$\sum_{i=1}^m 2^{m-i} x_{i,j} \geq \sum_{i=1}^m 2^{m-i} x_{i,j+1}, \quad \forall j \in \{1, \dots, n-1\} \quad (1)$$

As the 2^{m-i} term might cause numerical intractability, alternative inequalities featuring binary coefficients can be used, at the expense of losing the full symmetry-breaking property. An option is to use *column inequalities* introduced in [13]:

$$\sum_{k=1}^i x_{k,j} \geq x_{i,j+1}, \quad \forall j \in \{1, \dots, n-1\} \quad (2)$$

Another option is to use a partial-symmetry-breaking form of Friedman inequalities, as in [10, 18]:

$$\sum_{i=1}^m x_{i,j} \geq \sum_{i=1}^m x_{i,j+1}, \quad \forall j \in \{1, \dots, n-1\} \quad (3)$$

The latter inequalities enforce that the total number of ones in each column is non-increasing. Note that this does not guarantee lexicographically non increasing columns for the representatives.

For the particular case of packing (resp. partition) problems, *i.e.*, problems whose solution matrix features at most (resp. exactly) one 1-entry in each row, a class of full symmetry-breaking

inequalities is introduced in [13]. These inequalities lead to a complete linear description of two special cases of full orbitopes: the packing (resp. partitioning) orbitope, *i.e.*, the convex hull of all $m \times n$ binary matrices with lexicographically non-increasing columns and with at most (resp. exactly) one 1-entry per row. For the full orbitope, an $O(mn^3)$ extended formulation is given in [11]. To the best of our knowledge, it has never been used in practice to handle symmetries. A complete linear description of the full orbitope in the x variable space seems hard to reach [19]. For the full orbitope restricted to 2-column matrices, a complete linear description in the x space is available [19].

The Unit Commitment Problem (UCP) has demonstrated to be a good candidate to apply symmetry-breaking techniques [22, 18, 15, 4]. The Min-up/min-down Unit Commitment Problem (MUCP) is to find a minimum-cost power production plan on a discrete time horizon for a set of production units. At each time period, the total production has to meet a forecast demand. Each unit must satisfy minimum up-time and down-time constraints besides featuring production and start-up costs. In real-world UCP, some more technical constraints have also to be taken into account, such as ramp constraints limiting the increase and decrease in generated power at each time period. Classical formulations of the MUCP feature two sets of binary variables: up variables x and start-up variables u . In practical instances, there are several sets of identical units, *i.e.*, units with identical characteristics, which induce symmetries. Indeed, assuming a solution is expressed as a matrix where column j corresponds to the up/down trajectory of unit j over the time horizon, then any permutation of columns corresponding to identical units leads to another solution with same cost. Moreover, in some subproblems, there exist symmetries not contained in the symmetry group of the original problem. Indeed, units which are ready to start (resp. shut down) at a given time t can permute their plans from time t to T .

We propose a general framework to build full symmetry-breaking inequalities in order to handle sub-symmetries arising from solution subsets whose symmetry groups contain the symmetric group acting on some sub-columns. One additional variable per subset Q considered may be needed in these inequalities, depending whether variables x are sufficient to indicate that “ x belongs to subset Q ”. The proposed framework is applied to derive such inequalities when the symmetry group is the symmetric group \mathfrak{S}_n acting on the columns. It is also applied to derive inequalities breaking both symmetries and sub-symmetries in the MUCP. We present experimental results comparing these sub-symmetry-breaking inequalities to state-of-the-art symmetry-breaking formulations, such as the MUCP formulation featuring inequality (3) [18], aggregated (x, u) or aggregated interval MUCP formulations [15]. When the MUCP is considered, the integer decomposition property holds for the (x, u) formulation and thus efficient aggregation techniques apply [15]. When the ramp-constrained MUCP is considered, the integer decomposition property does not hold anymore for the (x, u) MUCP formulation, there the corresponding aggregated solutions can no longer be disaggregated. We show that our inequalities outperform all above mentioned formulations in the ramp-constrained case.

In Section 2, we describe the framework. In Section 3, an application to the symmetric group case is presented. In Section 4, the framework is applied to derive sub-symmetry-breaking inequalities dedicated to the MUCP. Experimental results on MUCP instances (with or without ramp constraints) are presented in Section 5.

2 Sub-symmetry-breaking inequalities

For a given solution subset Q , the symmetry group \mathcal{G}_Q of the corresponding subproblem is different from \mathcal{G} and may contain symmetries undetected in \mathcal{G} . In practice it is too expensive to compute the symmetry group for every subset $Q \subset \mathcal{X}$. However for many problems, symmetries of \mathcal{G} can be deduced from the problem's structure, and so can symmetries of \mathcal{G}_Q , for some particular solution subsets Q . In this case, symmetries of \mathcal{G}_Q are a priori known, and thus do not need to be computed. Such symmetries may be handled together with symmetries of \mathcal{G} . In this section, we introduce sub-symmetry-breaking inequalities designed to simultaneously handle symmetries and sub-symmetries in symmetric groups.

First, we provide an overview of sub-symmetries in integer programming.

2.1 Background on sub-symmetries

The sub-symmetry concepts introduced in [4] are briefly recalled in this section.

Consider a subset $Q \subset \mathcal{X}$ of solutions of (ILP) . The sub-symmetry group \mathcal{G}_Q relative to subset Q is defined as the symmetry group of subproblem $\min\{cx \mid x \in Q\}$. Permutations in sub-symmetry group \mathcal{G}_Q are referred to as *sub-symmetries*.

Let $\{Q_s \subset \mathcal{X}, s \in \{1, \dots, q\}\}$ be a set of solution subsets. To each $Q_s, s \in \{1, \dots, q\}$, corresponds a sub-symmetry group \mathcal{G}_{Q_s} . Let $O_k^s, k \in \{1, \dots, o_s\}$, be the orbits defined by \mathcal{G}_{Q_s} on subset $Q_s, s \in \{1, \dots, q\}$, and $\mathcal{O} = \{O_k^s, k \in \{1, \dots, o_s\}, s \in \{1, \dots, q\}\}$.

For given $x \in \mathcal{P}(m, n)$, let us define $\mathcal{G}(x) = \bigcup_{Q_s \ni x} \mathcal{G}_{Q_s}$, the set of all permutations π in $\bigcup_{s=1}^q \mathcal{G}_{Q_s}$ such that π can be applied to X .

Matrix x' is said to be in relation with $x \in \mathcal{P}(m, n)$ if there exist $r \in \mathbb{N}$ and permutations π_1, \dots, π_r such that $\pi_k \in \mathcal{G}(\pi_{k-1} \dots \pi_1(x)), \forall k \in \{1, \dots, r\}$, and $x' = \pi_1 \pi_2 \dots \pi_r(x)$. The *generalized orbit* \mathbb{O} of x with respect to $\{Q_s, s \in \{1, \dots, q\}\}$ is thus the set of all x' in relation with x . By definition, for any generalized orbit \mathbb{O} , there exist orbits $\sigma_1, \dots, \sigma_p \in \mathcal{O}$ such that $\mathbb{O} = \bigcup_{s=1}^p \sigma_s$. The set of representatives $\{\rho(\sigma), \sigma \in \mathcal{O}\}$ is said to be *orbit-compatible* if for any generalized orbit $\mathbb{O} = \bigcup_{s=1}^p \sigma_s, \sigma_1, \dots, \sigma_p \in \mathcal{O}$, there exists j such that $\rho(\sigma_j) = \rho(\sigma_s)$ for all i such that $\rho(\sigma_j) \in \sigma_s$. Such a solution $\rho(\sigma_j)$ is said to be a *generalized representative* of \mathbb{O} .

Given $x \in \mathcal{X}$ and sets $R \subset \{1, \dots, m\}$ and $C \subset \{1, \dots, n\}$, we consider submatrix (R, C) of x , denoted by $x(R, C)$, obtained by considering columns C of x on rows R only. Symmetry group \mathcal{G}_Q is the *sub-symmetric group* with respect to (R, C) if it is the set of all permutations of the columns of $x(R, C)$. If \mathcal{G}_Q is the sub-symmetric group with respect to (R, C) then subset Q is said to be *sub-symmetric* with respect to (R, C) .

Consider a set \mathbb{S} of solution subsets $Q_s, s \in \{1, \dots, q\}$, such that each subset $Q_s, s \in \{1, \dots, q\}$, is sub-symmetric with respect to (R_s, C_s) . For each orbit $O_k^s, k \in \{1, \dots, o_s\}$ of $\mathcal{G}_{Q_s}, s \in \{1, \dots, q\}$, its representative $x_k^s \in O_k^s$ is chosen to be such that submatrix $x_k^s(R_s, C_s)$ is lexicographically maximal, *i.e.*, its columns are lexicographically non-increasing. Such x_k^s is said to be the *lex-max* of orbit O_k^s with respect to (R_s, C_s) .

Property 1 ([4]). *The set of representatives $\{x_k^s, k \in \{1, \dots, o_s\}, s \in \{1, \dots, q\}\}$ is orbit-compatible.*

The *full sub-orbitope* $\mathcal{P}_{sub}(\mathbb{S})$ associated to \mathbb{S} is the convex hull of binary matrices x such that for each $s \in \{1, \dots, q\}$, if $x \in Q_s$ then the columns of $x(R_s, C_s)$ are lexicographically non-increasing.

2.2 Definition and validity

Consider a set \mathbb{S} of solution subsets Q_s , $s \in \{1, \dots, q\}$, such that each subset Q_s , $s \in \{1, \dots, q\}$, is sub-symmetric with respect to (R_s, C_s) . Consider integer variable z_s , $s \in \{1, \dots, q\}$, such that $z_s = 0$ if variable $x \in Q_s$, and such that $z_s \geq 1$ if $x \notin Q_s$. For any $x \in \mathcal{X}$, function Z associates x to a vector $Z(x)$ such that z_s , $s \in \{1, \dots, q\}$, is the s^{th} component of $Z(x)$ denoted by $Z_s(x)$.

Note that in many cases, function Z is linear, *i.e.*, each integer variable z_s is a linear expression of variables x . In such cases, no additional variable z_s is needed. In some cases where function Z is not linear, variable z_s can be linearly expressed from variables x using only a few additional inequalities or integer variables.

Given $c, c' \in C_s$ such that $c < c'$, the *sub-symmetry-breaking inequality*, denoted by $(Q_s(c, c'))$, is defined as follows.

$$x_{r_1, c'} \leq z_s + x_{r_1, c} \quad \text{where } r_1 = \min(R_s) \quad (4)$$

For each orbit O_k^s , $k \in \{1, \dots, o_s\}$, of \mathcal{G}_{Q_s} , $s \in \{1, \dots, q\}$, the chosen representative is the lex-max of orbit O_k^s with respect to (R_s, C_s) . Then by Property 1, this set of representatives is orbit-compatible. In particular, solution set \mathcal{X} can be restricted to the set of representatives by considering its intersection with the full sub-orbitope $\mathcal{P}_{\text{sub}}(\mathbb{S})$. If $x \in Q_s$, inequality $(Q_s(c, c'))$ enforces that the first row of submatrix $x(R_s, C_s)$ is lexicographically non-increasing, hence the following result.

Lemma 1 (Validity). *If $x \in \mathcal{P}_{\text{sub}}(\mathbb{S})$, then $(x, Z(x))$ satisfies inequality $(Q_s(c, c'))$ for each $s \in \{1, \dots, q\}$ and $c, c' \in C_s$ such that $c < c'$.*

Note that an inequality similar to (4) applied to a row of R_s distinct from r_1 may not be valid when used alongside with (4), as shown in Example 1.

Example 1. *Let $\mathbb{S} = \{Q_1\}$, $q = 1$, where subset Q_1 is as follows.*

$$Q_1 = \left\{ x \in \mathcal{P}(4, 3) \cap \mathcal{X} \mid \sum_{c=1}^3 x_{2,c} = 3 \right\}$$

*Let us suppose the symmetry group of Q_1 is the sub-symmetric group with respect to submatrix $(\{3, 4\}, \{1, 2, 3\})$. Variable z_1 can be defined using equality $z_1 = 3 - \sum_{c=1}^3 x_{2,c}$. Note that $z_1 = Z_1(x) = 0$ when $\sum_{c=1}^3 x_{2,c} = 3$, *i.e.*, $x \in Q_1$, and is positive otherwise. Here the first row in R_1 is $r_1 = \min(R_1) = 3$, thus given $c, c' \in \{1, 2, 3\}$, $c < c'$, inequality $(Q_1(c, c'))$ is as follows*

$$x_{3, c'} \leq \left(3 - \sum_{j=1}^3 x_{2, j} \right) + x_{3, c} \quad (5)$$

*Inequality $(Q_1(c, c'))$ enforces that row 3 of a solution matrix x is lexicographically ordered, *i.e.*, $x_{3,1} \geq x_{3,2} \geq x_{3,3}$, whenever $\sum_{c=1}^3 x_{2,c} = 3$.*

Now consider solutions $x^1, x^2 \in Q_1$:

$$x^1 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix} \quad \text{and} \quad x^2 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

Inequality (5) cuts off solution x^2 from the feasible set. Inequality (6) corresponds to inequality (4) applied to row 4:

$$x_{4,c'} \leq \left(3 - \sum_{j=1}^3 x_{2,j} \right) + x_{4,c} \quad (6)$$

Inequality (6) would cut off x^1 . This shows that inequalities (5) and (6) cannot be used simultaneously.

Note that in the general case, inequalities (4) may only be partial-symmetry-breaking. Indeed, for given $s \in \{1, \dots, q\}$ and $c, c' \in C_s$ such that $c < c'$, inequality $(Q_s(c, c'))$ only enforces that the first row of submatrix $x(R_s, C_s)$ is lexicographically non-increasing when $x \in Q_s$. In the case when $x_{r_1, c'} < x_{r_1, c}$, then sub-columns $x(R_s, \{c'\}) \prec x(R_s, \{c\})$. Otherwise, when $x_{r_1, c'} = x_{r_1, c}$, inequality (4) is not sufficient to select the lexmax representatives.

To enforce a lexicographical order, subsequent rows of submatrix $x(R_s, C_s)$ should be considered until a tie-break row is found. It is shown in the next section that inequalities $(Q_s(c, c'))$ for all $s \in \{1, \dots, q\}$ and $c < c' \in C_s$ enforce that $x \in \mathcal{P}_{sub}(\mathbb{S})$ provided a tie-break condition on set \mathbb{S} is fulfilled.

2.3 Full symmetry-breaking sufficient condition

In this section, we introduce a condition for inequalities (4) to be full symmetry-breaking.

For each $s \in \{1, \dots, q\}$, consider $R_s = \{r_1^s, \dots, r_{|R_s|}^s\}$ and $C_s = \{c_1^s, \dots, c_{|C_s|}^s\}$, where $r_1^s < \dots < r_{|R_s|}^s$ and $c_1^s < \dots < c_{|C_s|}^s$. For given $s \in \{1, \dots, q\}$ and any two columns $c_{l-1}^s, c_l^s \in C_s$, if there is a solution $x \in Q_s$ such that columns c_{l-1}^s and c_l^s are equal from row r_1^s to row r_{k-1}^s , it must be ensured that row r_k^s is lexicographically non increasing, *i.e.*, $x_{r_k^s, c_{l-1}^s} \leq x_{r_k^s, c_l^s}$. The key idea is to exhibit another set $Q_p \in \mathbb{S}$ for quartet (Q_s, k, l, x) , such that Q_p contains x and is sub-symmetric with respect to (R_p, C_p) , where the first row of R_p is r_k^s and C_p contains columns c_{l-1}^s and c_l^s . Then inequality $(Q_p(c_{l-1}^s, c_l^s))$ will ensure that $x_{r_k^s, c_{l-1}^s} \geq x_{r_k^s, c_l^s}$. For each quartet (Q_s, k, l, x) , the existence of such a subset Q_p in \mathbb{S} will be ensured by tie-break condition (C), defined as follows:

$$(C) \quad \begin{cases} \forall s \in \{1, \dots, q\}, \forall k \in \{2, \dots, |R_s|\}, \forall l \in \{2, \dots, |C_s|\} \\ \text{If } x \in Q_s \text{ such that } x_{r_{k'}^s, c_{l-1}^s} = x_{r_{k'}^s, c_l^s}, \forall k' \in \{1, \dots, k-1\}, \\ \text{then there exists } p \in \{1, \dots, q\} \text{ such that } x \in Q_p, C_p \supseteq \{c_{l-1}^s, c_l^s\} \text{ and } r_k^s = \min(R_p) \end{cases}$$

If condition (C) holds, inequalities $(Q_s(c_{l-1}^s, c_l^s))$, $\forall s \in \{1, \dots, q\}, \forall l \in \{2, \dots, |C_s|\}$ exactly restrict the solution set to the representative set $\mathcal{X} \cap \mathcal{P}_{sub}(\mathbb{S})$. They are therefore full symmetry breaking, with respect to the sub-symmetries defined by \mathbb{S} . This gives the idea of the proof for the following theorem.

Theorem 1. *If condition (C) holds, then:*

- (i) $(x, Z(x))$ satisfies $(Q_s(c_{l-1}^s, c_l^s))$, $\forall s \in \{1, \dots, q\}, \forall l \in \{2, \dots, |C_s|\}$
- (ii) $x \in \mathcal{P}_{sub}(\mathbb{S})$

are equivalent.

For general set \mathbb{S} , condition (C) may not hold. Fortunately, it will be shown that we can construct from \mathbb{S} another set $\tilde{\mathbb{S}}$ satisfying (C) and such that $\mathcal{P}_{sub}(\tilde{\mathbb{S}}) = \mathcal{P}_{sub}(\mathbb{S})$.

The idea is to divide each Q_s , $s \in \{1, \dots, q\}$ in smaller subsets such that for each row $r_k^s \in R_s$ and each column $c_l^s \in C_s$, there is a subset Q , which is sub-symmetric with respect to $(R, C) = (\{r_k^s, \dots, r_{|R_s|}^s\}, \{c_{l-1}^s, c_l^s\})$.

Set $\tilde{\mathbb{S}}$ is defined as follows.

$$\tilde{\mathbb{S}} = \left\{ \tilde{Q}_s(k, l) \mid s \in \{1, \dots, q\}, k \in \{1, \dots, |R_s|\}, l \in \{2, \dots, |C_s|\} \right\}$$

where for each $s \in \{1, \dots, q\}$, for each $l \in \{2, \dots, |C_s|\}$, for each $k \in \{1, \dots, |R_s|\}$,

$$\tilde{Q}_s(k, l) = \left\{ x \in Q_s \mid x_{r, c_{l-1}^s} = x_{r, c_l^s}, \forall r \in \{r_1^s, \dots, r_{k-1}^s\} \right\}$$

Note that for solution $x \in Q_s$ such that columns c_{l-1}^s and c_l^s are equal from row r_1^s to row r_{k-1}^s , the set exhibited for quartet (Q_s, k, l, x) is $\tilde{Q}_s(k, l)$. Note also that $\tilde{Q}_s(1, l) = Q_s$, $l \in \{2, \dots, |C_s|\}$.

We thus have the following result:

Lemma 2. *Set $\tilde{\mathbb{S}}$ satisfies (C) and is such that $\mathcal{P}_{sub}(\tilde{\mathbb{S}}) = \mathcal{P}_{sub}(\mathbb{S})$.*

Proof. The symmetry group of $\tilde{Q}_s(k, l)$ is the sub-symmetric group with respect to $(R, C) = (\{r_k^s, \dots, r_{|R_s|}^s\}, \{c_{l-1}^s, c_l^s\})$. Thus if some solution $x \in Q_s$ is such that columns c_{l-1}^s and c_l^s are equal from row r_1^s to row r_{k-1}^s , then subset $\tilde{Q}_s(k, l)$ contains x and is such that $C \supseteq \{c_{l-1}^s, c_l^s\}$ and $\min(R) = r_k^s$. Condition (C) is therefore satisfied by $\tilde{\mathbb{S}}$. It can be readily checked that the full sub-orbitopes defined by $\tilde{\mathbb{S}}$ and \mathbb{S} are the same. \square

It follows, from Theorem 1, that inequalities $(Q(c, c'))$, $c < c' \in C$, $Q \in \tilde{\mathbb{S}}$ are full symmetry-breaking with respect to the sub-symmetries defined by \mathbb{S} .

Corollary 1. *If for each $Q \in \tilde{\mathbb{S}}$, $(x, Z(x))$ satisfies inequality $(Q(c, c'))$, $\forall c < c' \in C$, then $x \in \mathcal{P}_{sub}(\mathbb{S})$.*

We can then consider $\tilde{\mathbb{S}}$ instead of \mathbb{S} . This implies to add one inequality at least (resp. one variable at most), per subset $Q \in \tilde{\mathbb{S}}$, i.e., $O(qmn)$ inequalities at least (resp. variables at most).

Example. Referring to Example 1, $\tilde{\mathbb{S}} = \{\tilde{Q}_1(1, l), \tilde{Q}_1(2, l), l \in \{2, 3\}\}$. For each $l \in \{2, 3\}$, $\tilde{Q}_1(1, l) = Q_1$ as for any s , $\tilde{Q}_s(k, l) = Q_s$ whenever $k = 1$. We also have $\tilde{Q}_1(2, l) = \{x \in Q_1 \mid x_{3, l-1} = x_{3, l}\}$. For each $l \in \{2, 3\}$, \tilde{z}_l associated to subset $\tilde{Q}_1(2, l)$ can be expressed as follows: $\tilde{z}_l = 2z_1 + (x_{3, l-1} - x_{3, l})$. Indeed, when $z_1 = 0$, inequality (5) becomes $x_{3, l-1} \leq x_{3, l}$. Thus, $\tilde{z}_l = 0$ if $x_{3, l-1} = x_{3, l}$ and $z_1 \geq 1$ otherwise. When $z_1 = 1$, $\tilde{z}_l \geq 1$. Hence the following inequalities are full symmetry-breaking:

$$\begin{aligned} x_{3, l-1} &\leq \left(3 - \sum_{j=1}^3 x_{2, j}\right) + x_{3, l} & \forall l \in \{2, 3\} \\ x_{4, l-1} &\leq \left(6 + x_{3, l-1} - x_{3, l} - 2 \sum_{j=1}^3 x_{2, j}\right) + x_{4, l} & \forall l \in \{2, 3\} \end{aligned}$$

In Sections 3 and 4, inequalities (4) are built in a more straightforward way, in the sense that set \mathbb{S} already satisfies condition (C) in the two applications studied.

3 Application to the symmetric group case

In this section, we apply the framework of Section 2 to any problem whose symmetry group \mathcal{G} is the symmetric group \mathfrak{S}_n acting on the columns. The collection $\mathbb{S}_{\mathfrak{S}}$ of subsets considered will lead to inequalities restricting any solution $x \in \mathcal{X}$ to be in the full orbitope. These inequalities feature variables z which can be explicitly expressed from x with $O(mn)$ linear inequalities. Here, the sub-symmetries considered are restrictions of symmetries' actions to solution subsets.

A complete linear description of the 2-column full orbitope, featuring additional integer variables, is proposed in [19]. In the general n -column case, we show that these inequalities can also be derived using the framework described in Section 2, and can be used as full symmetry-breaking inequalities.

We consider $\mathbb{S}_{\mathfrak{S}} = \{Q_{i,j}, i \in \{0\} \cup \{1, \dots, m-1\}, j \in \{2, \dots, n\}\}$, where

$$Q_{i,j} = \left\{ x \in \mathcal{X} \mid x_{i',j-1} = x_{i',j} \ \forall i' \in \{1, \dots, i\} \right\}.$$

Subset $Q_{i,j}$ is the set of feasible solutions such that columns $j-1$ and j are equal from row 1 to row i . Note that $Q_{0,j} = \mathcal{X}$. The symmetry group of $Q_{i,j}$ is then the sub-symmetric group with respect to $(R_i, \{j-1, j\})$ where $R_i = \{i+1, \dots, m\}$. It can be readily checked that in this case, \mathbb{S} already satisfies condition (C).

Let variable $z_{i,j}$ be such that $z_{i,j} = 0$ if $x \in Q_{i,j}$ and 1 otherwise. Note that for all $j \in \{2, \dots, n\}$, $Q_{0,j} = \mathcal{X}$, thus $z_{0,j} = 0, \forall x \in \mathcal{X}$. Note also that $\mathcal{X} \cap \mathcal{P}_{sub}(\mathbb{S}_{\mathfrak{S}})$ is a subset of the full orbitope.

Thus, given that the columns of any $x \in \mathcal{X} \cap \mathcal{P}_{sub}(\mathbb{S}_{\mathfrak{S}})$ are in a non-increasing lexicographical order, function Z is such that $Z(x) = z$, where z satisfies the following linear inequalities.

$$\begin{cases} z_{1,j-1} = x_{1,j-1} - x_{1,j} & \forall j \in \{2, \dots, n\} & (7a) \\ z_{i,j-1} \leq z_{i-1,j-1} + x_{i,j-1} & \forall i \in \{2, \dots, m\}, j \in \{2, \dots, n\} & (7b) \\ z_{i,j-1} + x_{i,j} \leq 1 + z_{i-1,j-1} & \forall i \in \{2, \dots, m\}, j \in \{2, \dots, n\} & (7c) \\ x_{i,j-1} \leq z_{i,j-1} + x_{i,j} & \forall i \in \{2, \dots, m\}, j \in \{2, \dots, n\} & (7d) \\ z_{i-1,j-1} \leq z_{i,j-1} & \forall i \in \{2, \dots, m\}, j \in \{2, \dots, n\} & (7e) \end{cases}$$

Constraint (7a) sets variable $z_{1,j-1}$ to 1 whenever columns $j-1$ and j are different and in a non-increasing lexicographical order on row 1, and to 0 when they are equal. Constraints (7b) and (7c) set variable $z_{i,j-1}$ to 0 when $z_{i-1,j-1} = 0$ and columns $j-1$ and j are equal on row i . Constraint (7d) sets variable $z_{i,j-1}$ to 1 if columns $j-1$ and j are different and in a non-increasing lexicographical order on row i . Constraint (7e) sets $z_{i,j-1}$ to 1 when variable $z_{i-1,j-1} = 1$, *i.e.*, when columns $j-1$ and j are different before row i .

For each $i \in \{0, \dots, m-1\}$ and $j \in \{2, \dots, n\}$ inequality (4) is inequality $(Q_{i,j}(j-1, j))$ as follows:

$$x_{i+1,j} \leq z_{i,j-1} + x_{i+1,j-1} \quad \forall i \in \{1, \dots, m\}, \forall j \in \{2, \dots, n\}$$

It ensures that if columns $j-1$ and j of x are equal from row 1 to i , then row $i+1$ is in a non-increasing lexicographical order.

Note that if $z_{i-1,j-1} - z_{i,j-1} = -1$ then necessarily $x_{i,j} = 0$. Thus inequality $((Q_{i,j}(j-1, j)))$ can be lifted to

$$x_{i,j} \leq (2z_{i-1,j-1} - z_{i,j-1}) + x_{i,j-1} \quad (8)$$

In the special case when $n = 2$, by replacing variable $z_{i,j}$ by $y_{i,j}$ where $z_{i,j} = 1 - \sum_{i'=1}^i y_{i',j}$, for each $i \in \{1, \dots, m\}$, $j \in \{1, 2\}$, inequalities (7a)–(8) yield the complete linear description of the 2-column full orbitope proposed in [19].

In the general n -column case, inequalities (7a)–(8) are still full symmetry-breaking (by Theorem 1), and then can be used in practice to restrict the feasible set to any full orbitope. In this case, $O(mn)$ additional variables and constraints are needed.

4 Application to the Unit Commitment Problem

The framework of Section 2 is now applied to the Unit Commitment Problem, which features many sub-symmetries non detected by the symmetry group \mathcal{G} .

Given a discrete time horizon $\mathcal{T} = \{1, \dots, T\}$, a demand for electric power D_t is to be met at each time period $t \in \mathcal{T}$. Power is provided by a set \mathcal{N} of n production units. At each time period, unit $j \in \mathcal{N}$ is either down or up, and in the latter case, its production is within $[P_{min}^j, P_{max}^j]$. Each unit must satisfy minimum up-time (resp. down-time) constraints, *i.e.*, it must remain up (resp. down) during at least L^j (resp. ℓ^j) periods after start up (resp. shut down). Each unit j also features three different costs: a fixed cost c_f^j , incurred each time period the unit is up; a start-up cost c_0^j , incurred each time the unit starts up; and a cost c_p^j proportional to its production. The Min-up/min-down Unit Commitment Problem (MUCP) is to find a production plan minimizing the total cost while satisfying the demand and the minimum up and down time constraints. The MUCP is strongly NP-hard [3].

In the real-world Unit Commitment Problem (UCP), some more technical constraints have also to be taken into account, such as ramp constraints or reserve requirement constraints, and the start-up costs are an exponential function of the unit downtime. From a combinatorial point of view, the MUCP is the core structure of the UCP. In this section, we study the MUCP with and without ramp constraints.

For each unit $j \in \mathcal{N}$ and time period $t \in \mathcal{T}$, let us consider three variables: $x_{t,j} \in \{0, 1\}$ indicates if unit j is up at time t ; $u_{t,j} \in \{0, 1\}$ whether unit j starts up at time t ; and $p_{t,j} \in \mathbb{R}$ is the quantity of power produced by unit j at time t . Without loss of generality we consider that $L^j, \ell^j \leq T$. Formulation $F(x, u)$ for the MUCP is as follows [25, 22, 2].

$$(F(x, u)) \quad \min_{x, u, p} \sum_{j=1}^n \sum_{t=1}^T c_f^j x_{t,j} + c_p^j p_{t,j} + c_0^j u_{t,j}$$

$$\text{s. t.} \quad \sum_{t'=t-L^j+1}^t u_{t',j} \leq x_{t,j} \quad \forall j \in \mathcal{N}, \forall t \in \{L^j, \dots, T\} \quad (9)$$

$$\sum_{t'=t-\ell^j+1}^t u_{t',j} \leq 1 - x_{t-\ell^j,j} \quad \forall j \in \mathcal{N}, \forall t \in \{\ell^j, \dots, T\} \quad (10)$$

$$u_{t,j} \geq x_{t,j} - x_{t-1,j} \quad \forall j \in \mathcal{N}, \forall t \in \{2, \dots, T\} \quad (11)$$

$$P_{min}^j x_{t,j} \leq p_{t,j} \leq P_{max}^j x_{t,j} \quad \forall j \in \mathcal{N}, \forall t \in \mathcal{T} \quad (12)$$

$$\sum_{j=1}^n p_{t,j} \geq D_t \quad \forall t \in \mathcal{T} \quad (13)$$

$$x_{t,j}, u_{t,j} \in \{0, 1\} \quad \forall j \in \mathcal{N}, \forall t \in \mathcal{T} \quad (14)$$

For convenience, we will also use variable $w_{t,j} = x_{t-1,j} - x_{t,j} + u_{t,j}$, indicating whether unit j shuts down at time t .

4.1 Symmetries and sub-symmetries in the UCP

Symmetries in the MUCP (and in the UCP) arise from the existence of groups of identical units, *i.e.*, units with identical characteristics $(P_{min}, P_{max}, L, \ell, c_f, c_0, c_p)$. The instance is partitioned into *types* $h \in \{1, \dots, H\}$ of n_h identical units. The unit set of type h is denoted by $\mathcal{N}_h = \{j_1^h, \dots, j_{n_h}^h\}$.

The solutions of the MUCP can be expressed as a series of binary matrices. For a given type h , we introduce matrix $x^h \in \mathcal{P}(T, n_h)$ such that entry $x_{t,k}^h$ corresponds to variable x_{t,j_k^h} , where j_k^h is the index of the k^{th} unit of type h , $k \in \{1, \dots, n_h\}$. Column j of matrix x^h corresponds to the up/down plan relative to the j^{th} unit of type h . Similarly, we introduce matrices u^h and p^h .

The set of all feasible $x = (x_{t,j})_{t \in \mathcal{T}, j \in \mathcal{N}}$ is denoted by \mathcal{X}_{MUCP} . Note that any solution matrix x (resp. u, p) can be partitioned in H matrices x^h (resp. u^h, p^h). Since all units of type h are identical, their production plans can be permuted, provided that the same permutation is applied to matrices x^h, u^h and p^h . Thus, the symmetry group \mathcal{G} contains the symmetric group \mathfrak{S}_{n_h} acting on the columns of x^h , for each unit type h . Consequently, for each type h , feasible solutions x^h can be restricted to be in the $T \times n_h$ full orbitope. As binary variables u are uniquely determined by variables x , breaking the symmetry on x variables will break the symmetry on u variables. Note that this restriction to the $T \times n_h$ full orbitope for each type h can possibly be done using inequalities from Section 3 featuring z variables.

There are also other sources of symmetry, arising from the possibility of permuting some sub-columns of matrices x^h . For example, consider two identical units. Suppose at some time period t , these two units are down and ready to start up. Then their plans after t can be permuted, even if they do not have the same up/down plan before t .

More precisely, a unit $j \in \mathcal{N}$ is *ready to start up* at time $t \in \{1, \dots, T\}$ if and only if $\forall t' \in \{t - \ell^j, \dots, t - 1\}, x_{t',j} = 0$. Similarly, a unit $j \in \mathcal{N}^k$ is *ready to shut down* at time $t \in \{1, \dots, T\}$ if and only if $\forall t' \in \{t - L^j, \dots, t - 1\}, x_{t',j} = 1$.

4.2 Sub-symmetry-breaking inequalities for the MUCP

For each time period $t \in \{1, \dots, T\}$ and any two consecutive units j_k^h, j_{k+1}^h of type h , $k \in \{1, \dots, n_h - 1\}$, consider the following subsets of \mathcal{X}_{MUCP} :

$$\overline{Q}_{k,h}^t = \{x \in \mathcal{X}_{MUCP} \mid x_{t',j} = 0, \forall t' \in \{t - \ell^h, \dots, t - 1\}, \forall j \in \{j_k^h, j_{k+1}^h\}\}$$

$$\underline{Q}_{k,h}^t = \{x \in \mathcal{X}_{MUCP} \mid x_{t',j} = 1, \forall t' \in \{t - L^h, \dots, t - 1\}, \forall j \in \{j_k^h, j_{k+1}^h\}\}$$

where ℓ^h (resp. L^h) is the minimum down (resp. up) time of units of type h .

Note that $\overline{Q}_{k,h}^t$ and $\underline{Q}_{k,h}^t$ are different from subsets $Q_{i,j}$ defined in Section 3. Actually, $Q_{t,j_{k+1}^h} \subset \overline{Q}_{k,h}^t$ and $Q_{t,j_k^h} \subset \underline{Q}_{k,h}^t$.

Let $\mathcal{G}_{\overline{Q}_{k,h}^t}$ and $\mathcal{G}_{\underline{Q}_{k,h}^t}$ be the sub-symmetry groups associated to $\overline{Q}_{k,h}^t$ and $\underline{Q}_{k,h}^t$, $t \in \{1, \dots, T\}$, $h \in \{1, \dots, H\}$, $k \in \{1, \dots, n_h - 1\}$. The sub-symmetries in $\mathcal{G}_{\overline{Q}_{k,h}^t}$ (resp. $\mathcal{G}_{\underline{Q}_{k,h}^t}$) are called *start-up sub-symmetries* (resp. *shut-down sub-symmetries*). Most of these sub-symmetries are not detected in the symmetry group of the MUCP.

Groups $\mathcal{G}_{\overline{Q}_{k,h}^t}$ and $\mathcal{G}_{\underline{Q}_{k,h}^t}$ contain the sub-symmetric group associated to the submatrix defined by rows and columns $(\{t, \dots, T\}, \{j_k^h, j_{k+1}^h\})$.

Applying results from Section 2, variables $\overline{z}_{k,h}^t$ and $\underline{z}_{k,h}^t$, indicating whether $x \in \overline{Q}_{k,h}^t$ and $x \in \underline{Q}_{k,h}^t$ respectively, can be directly derived from variables x and u :

$$\overline{z}_{k,h}^t = x_{t-\ell^h,j'} + \sum_{t'=t-\ell^h+1}^{t-1} u_{t',j'} + x_{t-\ell^h,j} + \sum_{t'=t-\ell^h+1}^{t-1} u_{t',j}$$

$$\underline{z}_{k,h}^t = 1 - x_{t-L^h,j'} + \sum_{t'=t-L^h+1}^{t-1} w_{t',j'} + 1 - x_{t-L^h,j} + \sum_{t'=t-L^h+1}^{t-1} w_{t',j}$$

where $j = j_k^h$ and $j' = j_{k+1}^h$ for sake of clarity.

Consider $\mathbb{S}_{MUCP} = \{\overline{Q}_{k,h}^t, \underline{Q}_{k,h}^t, t \in \{1, \dots, T\}, h \in \{1, \dots, H\}, k \in \{1, \dots, n_h - 1\}\}$. In this case, set \mathbb{S} directly satisfies condition \mathcal{C} .

For each $h \in \{1, \dots, H\}$, $k \in \{1, \dots, n_h - 1\}$ and $t \in \{1, \dots, T\}$, inequalities $(\overline{Q}_{k,h}^t(j, j'))$ and $(\underline{Q}_{k,h}^t(j, j'))$, where $j = j_k^h$ and $j' = j_{k+1}^h$, are as follows.

$$x_{t,j'} \leq \left[x_{t-\ell^h,j'} + \sum_{t'=t-\ell^h+1}^{t-1} u_{t',j'} \right] + \left[x_{t-\ell^h,j} + \sum_{t'=t-\ell^h+1}^{t-1} u_{t',j} \right] + x_{t,j}$$

$$x_{t,j'} \leq \left[1 - x_{t-L^h,j'} + \sum_{t'=t-L^h+1}^{t-1} w_{t',j'} \right] + \left[1 - x_{t-L^h,j} + \sum_{t'=t-L^h+1}^{t-1} w_{t',j} \right] + x_{t,j}$$

Strengthening symmetry-breaking inequalities Inequalities $(\overline{Q}_{k,h}^t(j,j'))$ and $(\underline{Q}_{k,h}^t(j,j'))$ can be further strengthened, using the relationship between variables x and u .

First note that by definition of variables w :

$$x_{t,j'} - \left[x_{t-\ell^h,j'} + \sum_{t'=t-\ell^h+1}^{t-1} u_{t',j'} \right] = u_{t,j'} - \sum_{t'=t-\ell^h+1}^t w_{t',j'}$$

$$x_{t,j} + \left[1 - x_{t-L^h,j} + \sum_{t'=t-L^h+1}^{t-1} w_{t',j} \right] = -w_{t,j} + 1 + \sum_{t'=t-L^h+1}^t u_{t',j}$$

As if $u_{t,j'} = 1$ (resp. $w_{t,j} = 1$), then $\sum_{t'=t-\ell^h+1}^t w_{t',j'} = 0$ (resp. $\sum_{t'=t-L^h+1}^t u_{t',j} = 0$), the following *Start-Up-Ready* and *Shut-Down-Ready* inequalities are valid and stronger than inequalities $(\overline{Q}_{k,h}^t(j,j'))$ and $(\underline{Q}_{k,h}^t(j,j'))$.

$$u_{t,j'} \leq \left[x_{t-\ell^h,j} + \sum_{t'=t-\ell^h+1}^{t-1} u_{t',j} \right] + x_{t,j} \quad (15)$$

$$w_{t,j} \leq \left[1 - x_{t-L^h,j'} + \sum_{t'=t-L^h+1}^{t-1} w_{t',j'} \right] + 1 - x_{t,j'} \quad (16)$$

Note that for any $h \in \{1, \dots, H\}$ and $k \in \{1, \dots, n_h - 1\}$, $\overline{Q}_{k,h}^1 = \underline{Q}_{k,h}^1 = \mathcal{X}_{MUCP}$. As condition (C) is satisfied by \mathbb{S}_{MUCP} , any $x = (x_1, \dots, x_H)$ satisfying inequalities (15) and (16) is such that x_h is in the $T \times n_h$ full orbitope, $h \in \{1, \dots, H\}$. Hence inequalities (15) and (16) ensure in particular that any solution x_h is in the full orbitope.

4.3 Sub-symmetry-breaking inequalities for the ramp-constrained MUCP

In the real-world UCP, each unit j must also feature *ramp-up* (resp. *ramp-down*) constraints, *i.e.*, the maximum increase (resp. decrease) in generated power from time period t to time period $t+1$ is RU^j (resp. RD^j). Moreover, if unit i starts up at time t (resp. shuts down at time $t+1$), its production at time t cannot be higher than SU^j (resp. SD^j).

For each unit $j \in \mathcal{N}$ and time period $t \in \{2, \dots, T\}$, ramp constraints can be formulated as follows:

$$p_{t,j} - p_{t-1,j} \leq RU^j x_{t-1,j} + SU^j u_{t,j} \quad (17)$$

$$p_{t-1,j} - p_{t,j} \leq RD^j x_{t,j} + SD^j w_{t,j} \quad (18)$$

The MUCP formulation including ramp constraints can be further strengthened with valid inequalities as proposed in [21, 24]. As the aim of this article is to compare symmetry-breaking techniques, we will only consider the classical MUCP formulation (9) – (14) with ramp-constraints (17) – (18), as done in [22, 14].

When ramp-constraints are considered, the symmetry group of set $\overline{Q}_{k,h}^t$ still contains the sub-symmetric group associated to the submatrix defined by rows and columns $(\{t, \dots, T\}, \{j_k^h, j_{k+1}^h\})$. Therefore, inequalities (15) can still be used.

However the symmetry group of set $\underline{Q}_{k,h}^t$ no longer contains the sub-symmetric group associated to the submatrix defined by rows and columns $(\{t, \dots, T\}, \{j_k^h, j_{k+1}^h\})$. Indeed, if two identical units have been up for at least L^h time periods at time $t - 1$, they may produce distinct power values at time $t - 1$ and thus, because of ramp constraints, their up/down trajectories from time t to T cannot be permuted. Therefore, inequalities (16) can no longer be used.

Note that when two identical ramp-constrained units are ready to shut down, there still exist some sub-symmetries that could be exploited. These sub-symmetries are more intricate because they depend, for example, on the quantity of power produced by both units, or on the time of their last start-up.

5 Experimental results

In this section, we compare various formulations for the MUCP with or without ramp constraints. Some symmetry-breaking techniques need to interfere with the branching process. These are typically implemented using a callback instruction which deeply affects the performance of commercial solvers like Cplex. Consequently in our computational comparison, we only consider symmetry-breaking techniques that do not require the use of a callback.

5.1 Aggregated formulations for the UCP

In [15], the authors propose to break symmetries of the UCP by aggregating variables corresponding to identical units. This method is shown to outperform existing symmetry-breaking inequalities.

• **Aggregated (x, u) formulation** In the case of the MUCP, variables x, u of formulation (9–14) are aggregated into variables $\tilde{x}_{t,h} = \sum_{j \in \mathcal{N}_h} x_{t,j} \in \{0, \dots, n_h\}$ (resp. $\tilde{u}_{t,h} = \sum_{j \in \mathcal{N}_h} u_{t,j} \in \{0, \dots, n_h\}$) indicating how many units of type h are up (resp. start up) at time t . Variables $\tilde{p}_{t,h} = \sum_{j \in \mathcal{N}_h} p_{t,j} \in \mathbb{R}$ is the total amount of power produced at time t by units of type h . Formulation (9–14) becomes

$$\begin{aligned} \text{A}(\tilde{x}, \tilde{u}) \quad & \min_{\tilde{x}, \tilde{u}, \tilde{p}} \sum_{h=1}^H \sum_{t=1}^T c_f^h \tilde{x}_{t,h} + c_p^j \tilde{p}_{t,h} + c_0^j \tilde{u}_{t,h} \\ \text{s. t.} \quad & \sum_{t'=t-L^h+1}^t \tilde{u}_{t',h} \leq \tilde{x}_{t,h} \quad \forall h \in \mathcal{N}_h, \forall t \in \{L^h, \dots, T\} \end{aligned} \quad (19)$$

$$\sum_{t'=t-\ell^h+1}^t \tilde{u}_{t',h} \leq n_h - \tilde{x}_{t-\ell^h,h} \quad \forall h \in \mathcal{N}_h, \forall t \in \{\ell^h, \dots, T\} \quad (20)$$

$$\tilde{u}_{t,h} \geq \tilde{x}_{t,h} - \tilde{x}_{t-1,h} \quad \forall h \in \mathcal{N}_h, \forall t \in \{2, \dots, T\} \quad (21)$$

$$P_{min}^h \tilde{x}_{t,h} \leq \tilde{p}_{t,h} \leq P_{max}^h \tilde{x}_{t,h} \quad \forall h \in \mathcal{N}_h, \forall t \in \mathcal{T} \quad (22)$$

$$\sum_{h=1}^H \tilde{p}_{t,h} \geq D_t \quad \forall t \in \mathcal{T} \quad (23)$$

$$\tilde{x}_{t,h}, \tilde{u}_{t,h} \in \{0, \dots, n_h\} \quad \forall h \in \mathcal{N}_h, \forall t \in \mathcal{T} \quad (24)$$

When aggregating variables corresponding to h identical units, one must ensure that the aggregated production plan, satisfying (19)–(22), can be disaggregated into h feasible production plans satisfying (9)–(12). Inequalities (9)–(12) have the *integer decomposition property* [1], *i.e.*, any integer solution $(\tilde{x}, \tilde{u}, \tilde{p})$ of formulation (19)–(24) can be disaggregated into an integer solution (x, u, p) of formulation (9)–(14). A disaggregation algorithm for the MUCP is proposed in [15].

When ramp constraints are considered in formulation (9)–(14), the integer decomposition property is lost. Examples of aggregated solutions which cannot be disaggregated are given in [15].

• **Aggregated interval formulation** As the integer decomposition property depends on the formulation considered, an interval-based formulation is introduced in [15] for the ramp-constrained MUCP. For each unit $j \in \mathcal{N}$, for each interval $\{t_0, \dots, t_1 - 1\}$ of size $t_1 - t_0 \geq L^j$, variable $y_j^{t_0, t_1} = 1$ if and only if unit j starts up at time t_0 , remains up on interval $\{t_0, \dots, t_1 - 1\}$ and shuts down at time t_1 . For each time period $t \in \mathcal{T}$, variable $p_{t,j}^{t_0, t_1}$ represents the quantity of power produced by unit j at time t if $y_j^{t_0, t_1} = 1$, and $p_{t,j}^{t_0, t_1} = 0$ otherwise. The formulation is as follows.

$$\min_{y, p} \sum_{j=1}^n \sum_{\{t_0, \dots, t_1-1\} \in \mathcal{Y}_j} c_{t_0, t_1}^j y_j^{t_0, t_1} + c_p^j \sum_{t=t_0}^{t_1-1} p_{t,j}^{t_0, t_1}$$

$$\text{s. t.} \quad A_j^{t_0, t_1} p_j^{t_0, t_1} \leq b_j^{t_0, t_1} y_j^{t_0, t_1} \quad \forall j \in \mathcal{N}, \forall \{t_0, \dots, t_1 - 1\} \in \mathcal{Y}_j \quad (25)$$

$$\sum y_j^{t_0, t_1} \leq 1 \quad \forall j \in \mathcal{N}, \forall t \in \mathcal{T} \quad (26)$$

$$\{t_0, \dots, t_1 - 1\} \in \mathcal{Y}_j$$

$$\text{s.t. } t \in \{t_0, \dots, t_1 + \ell^j\}$$

$$\sum_{j \in \mathcal{N}} \sum_{\{t_0, \dots, t_1-1\} \in \mathcal{Y}_j} p_{t,j}^{t_0, t_1} \geq D_t \quad \forall t \in \mathcal{T} \quad (27)$$

$$y_j^{t_0, t_1} \in \{0, 1\} \quad \forall j \in \mathcal{N}, \forall \{t_0, \dots, t_1 - 1\} \in \mathcal{Y}_j \quad (28)$$

where $\mathcal{Y}_j = \{\{t_0, \dots, t_1 - 1\} \in T \times T \mid t_1 - t_0 \geq L^j\}$ and where $\mathcal{P}_j^{t_0, t_1} = \{p_j^{t_0, t_1} \in \mathbb{R}_+^T \mid A_j^{t_0, t_1} p_j^{t_0, t_1} \leq b_j^{t_0, t_1}\}$ is the feasible production polytope of unit j , if unit j starts up at time t_0 , remains up on interval $\{t_0, \dots, t_1 - 1\}$ and shuts down at time t_1 . In the case of the ramp-constrained MUCP, $\mathcal{P}_j^{t_0, t_1}$ is defined by power limits (29) and ramp constraints (30)–(33):

$$P_{min}^j \leq p_{t,j}^{t_0, t_1} \leq P_{max}^j \quad (29)$$

$$p_{t,j}^{t_0, t_1} - p_{t-1,j}^{t_0, t_1} \leq RU^j \quad \forall t \in \{t_0 + 1, \dots, t_1 - 1\} \quad (30)$$

$$p_{t-1,j}^{t_0, t_1} - p_{t,j}^{t_0, t_1} \leq RD^j \quad \forall t \in \{t_0 + 1, \dots, t_1 - 1\} \quad (31)$$

$$p_{t_0,j}^{t_0, t_1} \leq SU^j \quad (32)$$

$$p_{t_1-1,j}^{t_0, t_1} \leq SD^j \quad (33)$$

Inequalities (25)–(26) have the integer decomposition property, thus variables $y_j^{t_0, t_1}$ (resp. $p_{t,j}^{t_0, t_1}$) can be aggregated into variables $\tilde{y}_h^{t_0, t_1} = \sum_{j \in \mathcal{N}_h} y_j^{t_0, t_1}$ and $\tilde{p}_{t,h}^{t_0, t_1} = \sum_{j \in \mathcal{N}_h} p_{t,j}^{t_0, t_1}$, leading to aggregated formulation $\text{Int}(\tilde{y})$:

$$\begin{aligned}
\text{Int}(\tilde{y}) \quad & \min_{\tilde{y}, \tilde{p}} \sum_{h=1}^H \sum_{\{t_0, \dots, t_1-1\} \in \mathcal{Y}_h} c_{t_0, t_1}^h \tilde{y}_h^{t_0, t_1} + c_p^h \sum_{t=t_0}^{t_1-1} \tilde{p}_{t, h}^{t_0, t_1} \\
\text{s. t.} \quad & A_h^{t_0, t_1} \tilde{p}_h^{t_0, t_1} \leq b_h^{t_0, t_1} \tilde{y}_h^{t_0, t_1} \quad \forall h \in \{1, \dots, H\}, \forall \{t_0, \dots, t_1-1\} \in \mathcal{Y}_h \quad (34) \\
& \sum \tilde{y}_j^{t_0, t_1} \leq n_h \quad \forall h \in \{1, \dots, H\}, \forall t \in \mathcal{T} \quad (35) \\
& \{t_0, \dots, t_1-1\} \in \mathcal{Y}_j \\
& \text{s.t. } t \in \{t_0, \dots, t_1 + \ell^j\} \\
& \sum_{h=1}^H \sum_{\{t_0, \dots, t_1-1\} \in \mathcal{Y}_h} \tilde{p}_{t, h}^{t_0, t_1} \geq D_t \quad \forall t \in \mathcal{T} \quad (36) \\
& \tilde{y}_h^{t_0, t_1} \in \{0, \dots, n_h\} \quad \forall h \in \{1, \dots, H\}, \forall \{t_0, \dots, t_1-1\} \in \mathcal{Y}_h \quad (37)
\end{aligned}$$

5.2 Experimental settings

In this section, we compare various symmetry-breaking formulations for the MUCP with or without ramp-constraints.

As shown in [22], neither Friedman inequalities (1) nor column inequalities (2) are competitive with respect to the classical UCP formulation when solved by Cplex.

On the opposite, the weaker form of Friedman inequality (3) has been shown in [18] to outperform Default Cplex.

Hence the following formulations for the MUCP are compared:

- $F(x, u)$: (x, u) -formulation (9)–(14)
- $A-(\tilde{x}, \tilde{u})$: Aggregated (\tilde{x}, \tilde{u}) -formulation (19)–(24) (only when disaggregation applies)
- $\text{Int}(\tilde{y})$: Aggregated interval formulation (34)–(37)
- $W(x, u)$: (x, u) -formulation (9)–(14) with weaker Friedman inequalities (3)
- $F(x, u, z)$: (x, u) -formulation (9)–(14) with variables z , inequalities (7a)–(7e) and sub-symmetry-breaking inequalities (8)
- $LF(x, u)$: (x, u) -formulation (9)–(14) with sub-symmetry-breaking inequalities (15)–(16).

Formulation $F(x, u, z)$ is obtained from the classical MUCP formulation $F(x, u)$ by a direct use of the inequalities given in Section 3. As seen in Section 4, taking into account sub-symmetries in the MUCP leads to formulation $LF(x, u)$ featuring lifted symmetry breaking-inequalities (15) and (16), namely Start-up-ready and Shut-down-ready inequalities, in place of inequalities (7a)–(8). Note that the start-up and shut-down sub-symmetries of the MUCP are not handled by formulations $F(x, u)$, $W(x, u)$ and $F(x, u, z)$.

Formulations $F(x, u)$, $W(x, u)$, $F(x, u, z)$ and $LF(x, u)$ feature $O(nT)$ variables while formulation $A-(\tilde{x}, \tilde{u})$ (resp. $\text{Int}(\tilde{y})$) features $O(HT)$ (resp. $O(T^2H)$) variables, where H is the number of groups of identical units.

For the ramp-constrained MUCP, inequalities (17)–(18) enforcing ramp-constraints are added to formulations $F(x, u)$, $W(x, u)$, $F(x, u, y)$ and $LF(x, u)$. Aggregated formulation $A-(\tilde{x}, \tilde{u})$ can no longer be used, as its solutions cannot be disaggregated [15]. Note also that in this context, Start-up-ready inequalities are adjoined to $LF(x, u)$, but Shut-down-ready inequalities cannot.

In formulation $\text{Int}(\tilde{y})$, the production limit constraint (29) is always included in inequalities (34) defining feasible productions. In the ramp-constrained case, the ramp constraints (30)–(33) are also included.

All experiments are performed using Cplex 12.8 C++ API on 8 threads of a PC with a 64 bit Intel Core i7-6700 processor running at 3.4GHz, and 32 GB of RAM memory. The UCP instances are solved until optimality (defined within 10^{-7} of relative optimality tolerance) or until the time limit of 3600 seconds is reached.

5.3 Instances

We generate MUCP instances as follows.

For each instance, we generate a “2-peak per day” type demand with a large variation between peak and off-peak values: during one day, the typical demand in energy has two peak periods, one in the morning and one in the evening. The amplitudes between peak and off-peak periods have similar characteristics to those in the dataset from [6].

We consider the parameters $(P_{min}, P_{max}, L, \ell, c_f, c_0, c_p)$ of each unit from the dataset presented in [6]. We draw a correlation matrix between these characteristics and define a possible range for each characteristic. In order to introduce symmetries in our instances, some units are randomly generated based on the parameters correlations and ranges. Each unit generated is duplicated d times, where d is randomly selected in $[1, \frac{n}{F}]$ in order to obtain a total of n units. The parameter F is called symmetry factor, and can vary from 2 to 4 depending on the value of n . Note that these instances are generated along the same lines as literature instances considered in [2], but with different F factors.

In order to determine which symmetry-breaking technique performs best with respect to the number of rows and columns of matrices in feasible set \mathcal{X} , we consider various instance sizes $n \in \{20, 30, 60\}$ and $T \in \{48, 60\}$, and various symmetry factors $F \in \{2, 3, 4\}$. For each size (n, T) and symmetry factor F , we generate a set of 20 instances. Symmetry factor $F = 4$ is not considered for instances with a small number n of units ($n = 20$ or 30), as it leads to very small sets of identical units.

Table 1 provides some statistics on the instances characteristics. For each instance, a group is a set of two or more units with the same characteristics. Each unit which has not been duplicated is a singleton. The first and second entries column-wise are the number of singletons and groups. The third entry is the average group size and the fourth entry is the maximum group size. Each entry row-wise corresponds to the average value obtained over 20 instances with same size (n, T) and same symmetry factor F .

The ramp-constrained MUCP instances considered are the same as in the non-ramp-constrained case, with additional ramp characteristics $RU^j = \frac{P_{max}^j - P_{min}^j}{3}$, $RD^j = \frac{P_{max}^j - P_{min}^j}{2}$ and $SU^j = SD^j = P_{min}^j$.

5.4 Results for the MUCP

Tables 2 and 3 provide, for each formulation and each group of 20 instances:

#opt:	Number of instances solved to optimality,
#nodes:	Average number of nodes,
gap:	Average optimality gap,
CPU time:	Average CPU time in seconds.

Size (n, T)	Sym. factor	Nb singl.	Nb groups	Av. group size	Group max. size
(20,48)	$F = 3$	1.25	4.90	3.96	5.75
	$F = 2$	0.75	3.20	6.45	8.75
(20,96)	$F = 3$	0.90	4.75	4.08	5.60
	$F = 2$	0.75	3.45	5.93	8.65
(30,48)	$F = 3$	1.10	5.35	5.51	9.45
	$F = 2$	0.25	3.85	8.30	12.60
(30,96)	$F = 3$	0.40	5.25	5.97	8.65
	$F = 2$	0.55	4.05	7.59	11.40
(60,48)	$F = 4$	0.80	7.70	7.86	13.20
	$F = 3$	0.55	5.80	10.90	17.80
	$F = 2$	0.20	4.75	13.90	23.80
(60,96)	$F = 4$	0.60	7.90	7.79	13.20
	$F = 3$	0.30	5.95	10.50	16.60
	$F = 2$	0.20	4.35	14.80	24.90

Table 1: Instance characteristics

Note that a sign “-” in the column entry corresponding to the CPU time means that no instance could be solved within the time limit.

It is clear from Table 2 that aggregated (x, u) formulation A- (\tilde{x}, \tilde{u}) outperforms by far all the other formulations. This could be explained by the reduced size of aggregated formulation A- (\tilde{x}, \tilde{u}) , but also by the good performance of Cplex on ILP featuring integer variables (with bounds greater than 1). This efficiency will certainly be preserved any time the integer decomposition property holds for an (x, u) formulation of the UCP. Aggregated interval formulation Int(\tilde{y}) is in average one or even two order of magnitude slower than $F(x, u)$, $F(x, u, z)$ and $LF(x, u)$. Formulations $F(x, u, z)$ and $W(x, u)$ are always outperformed by $F(x, u)$ and $LF(x, u)$. Formulations $F(x, u)$ and $LF(x, u)$ are quite comparable on $(n, T) = (20, 48)$ instances. Interestingly, on $(n, T) = (20, 96)$ instances, $LF(x, u)$ is better than $F(x, u)$. Otherwise, when n is larger (*i.e.*, $n \geq 30$), and when $T = 96$ or when $F = 2$, $F(x, u)$ outperforms $LF(x, u)$, probably because more sub-symmetry-breaking inequalities are added in this case. When the horizon size is smaller (*i.e.*, $T=48$) and when $F \in \{3, 4\}$, fewer sub-symmetry-breaking inequalities are needed, which allows formulation $LF(x, u)$ to outperform $F(x, u)$ for $n = 30$ and $n = 60$.

5.5 Results for the ramp-constrained MUCP

Recall that aggregated formulation A- (\tilde{x}, \tilde{u}) can no longer be used in this context.

Tables 4 and 5 provide, for each formulation and each group of 20 instances, the exact same column entries as those in Tables 2 and 3.

First note that the ramp constraints make the MUCP instances much harder to solve by Cplex in general, as the CPU times in Tables 4 and 5 are much larger than those in Tables 2 and 3. For example, the integrality gap is in average 10 times larger for ramp-constrained problems on $(n, T) = (60, 48)$ and $F = 2$ instances.

Formulation Int(\tilde{y}) is still the less efficient formulation. It does not solve to optimality any

Instances		Formulation	#opt	#nodes	gap (%)	CPU time
(20,48)	$F = 2$	$F(x, u)$	20	1271	0	2.6
		$A-(\tilde{x}, \tilde{u})$	20	0	0	0.2
		$\text{Int}(\tilde{y})$	16	205 667	0.005 02	781.6
		$W(x, u)$	20	4809	0	13.7
		$F(x, u, z)$	20	3838	0	23.2
		$LF(x, u)$	20	1915	0	6.6
	$F = 3$	$F(x, u)$	20	806	0	2.6
		$A-(\tilde{x}, \tilde{u})$	20	0	0	0.3
		$\text{Int}(\tilde{y})$	18	152 948	0.001 57	572.1
		$W(x, u)$	20	1600	0	4.4
		$F(x, u, z)$	20	683	0	6.7
		$LF(x, u)$	20	271	0	3.5
(20,96)	$F = 2$	$F(x, u)$	20	148 942	0	267.3
		$A-(\tilde{x}, \tilde{u})$	20	0	0	0.7
		$\text{Int}(\tilde{y})$	6	13 180	9.368 57	2977.5
		$W(x, u)$	18	110 644	0.000 15	459.6
		$F(x, u, z)$	18	118 877	0.000 18	497.8
		$LF(x, u)$	19	52 881	0.000 13	215.1
	$F = 3$	$F(x, u)$	18	29 418	0.032 71	376.7
		$A-(\tilde{x}, \tilde{u})$	20	2360	0	8.2
		$\text{Int}(\tilde{y})$	7	32 859	0.168 55	2574.8
		$W(x, u)$	19	79 864	0.000 61	357.1
		$F(x, u, z)$	18	39 694	0.000 92	458.3
		$LF(x, u)$	19	19 831	0.000 85	229.1
(30,48)	$F = 2$	$F(x, u)$	20	3207	0	5.5
		$A-(\tilde{x}, \tilde{u})$	20	0	0	0.2
		$\text{Int}(\tilde{y})$	13	615 979	0.002 11	1483.4
		$W(x, u)$	18	67 553	0.000 18	370.7
		$F(x, u, z)$	18	41 669	0.000 21	391.5
		$LF(x, u)$	20	3894	0	19.4
	$F = 3$	$F(x, u)$	19	157 903	0.000 01	189.1
		$A-(\tilde{x}, \tilde{u})$	20	6	0	0.4
		$\text{Int}(\tilde{y})$	13	267 844	0.007 72	1548.7
		$W(x, u)$	19	108 689	0.000 02	283.9
		$F(x, u, z)$	19	30 450	0	249.5
		$LF(x, u)$	20	6044	0	18.5
(30,96)	$F = 2$	$F(x, u)$	20	39 633	0	131.2
		$A-(\tilde{x}, \tilde{u})$	20	21	0	0.5
		$\text{Int}(\tilde{y})$	0	21 783	2.002 12	-
		$W(x, u)$	18	54 860	0.000 29	679.8
		$F(x, u, z)$	18	20 062	0.000 83	758.5
		$LF(x, u)$	19	17 040	0	299.5
	$F = 3$	$F(x, u)$	20	15 653	0	53.5
		$A-(\tilde{x}, \tilde{u})$	20	215	0	0.6
		$\text{Int}(\tilde{y})$	0 18	30 267	4.389 68	-
		$W(x, u)$	13	324 913	0.000 09	1400.7
		$F(x, u, z)$	17	123 943	0.000 03	907.1
		$LF(x, u)$	19	146 639	0.000 03	299.6

Table 2: Performance indicators relative to the comparison of six formulations for MUCP instances with symmetries and $n = 20, 30$

Instances		Formulation	#opt	#nodes	gap (%)	CPU time
(60,48)	$F = 2$	$F(x, u)$	19	60 498	0	317.3
		$A-(\tilde{x}, \tilde{u})$	20	69	0	0.2
		$\text{Int}(\tilde{y})$	8	231 718	0.005 47	2345.3
		$W(x, u)$	17	43 468	0.000 08	825.5
		$F(x, u, z)$	19	6886	0	1234.4
		$LF(x, u)$	18	20 969	0.000 07	727.6
	$F = 3$	$F(x, u)$	19	154 845	0.000 05	308.2
		$A-(\tilde{x}, \tilde{u})$	20	17	0	0.4
		$\text{Int}(\tilde{y})$	2	432 040	0.010 18	3287.8
		$W(x, u)$	17	82 309	0.000 07	602.5
		$F(x, u, z)$	19	20 455	0.000 05	674.7
		$LF(x, u)$	20	2362	0	87.3
	$F = 4$	$F(x, u)$	17	326 005	0.000 22	587.4
		$A-(\tilde{x}, \tilde{u})$	20	79	0	0.3
		$\text{Int}(\tilde{y})$	9	298 344	0.006 72	2188.4
		$W(x, u)$	20	107 657	0	349.9
		$F(x, u, z)$	18	52 481	0.000 08	893.3
		$LF(x, u)$	20	1222	0	32.0
(60,96)	$F = 2$	$F(x, u)$	17	186 561	0.000 13	732.7
		$A-(\tilde{x}, \tilde{u})$	20	0	0	0.3
		$\text{Int}(\tilde{y})$	2	52 352	0.080 25	3421.5
		$W(x, u)$	8	202 197	0.000 64	2443.1
		$F(x, u, z)$	7	25 805	0.000 56	2875.7
		$LF(x, u)$	13	147 301	0.000 27	1850.6
	$F = 3$	$F(x, u)$	12	831 214	0.000 44	1765.1
		$A-(\tilde{x}, \tilde{u})$	20	98	0	0.6
		$\text{Int}(\tilde{y})$	1	6201	17.591 09	3586.3
		$W(x, u)$	6	287 697	0.000 92	2603.2
		$F(x, u, z)$	5	86 252	0.002 64	3051.5
		$LF(x, u)$	9	457 416	0.000 66	2190.5
	$F = 4$	$F(x, u)$	16	297 595	0.000 20	906.0
		$A-(\tilde{x}, \tilde{u})$	20	39	0	0.9
		$\text{Int}(\tilde{y})$	1	6093	44.510 52	3426.2
		$W(x, u)$	7	417 944	0.000 44	2566.1
		$F(x, u, z)$	9	59 442	0.000 84	2498.6
		$LF(x, u)$	10	386 902	0.000 26	1902.3

Table 3: Performance indicators relative to the comparison of six formulations for MUCP instances with symmetries and $n = 60$

instance with $n > 20$ but one. Moreover, on $n = 30$ instances, and on $(n, T) = (60, 96)$ instances, the root node cannot be processed at all within the time limit for formulation $\text{Int}(\tilde{y})$; the number of nodes explored is 0 and the optimality gap is 100%.

Formulation $LF(x, u)$ is more efficient than all considered formulations. Except on $(n, T) = (20, 96)$ and $F = 2$ instances where $F(x, u, z)$ manages to solve to optimality two instances more than $LF(x, u)$, formulation $LF(x, u)$ is able to solve a larger number of instances to optimality than all considered formulations. In particular, it manages to solve to optimality two of the large-size instances (*i.e.*, $(n, T) = (60, 96)$), while other formulations do not reach optimality on any of these instances. Moreover, formulation $LF(x, u)$ solves 52 instances to optimality among the 80 instances with $n = 30$, while $F(x, u, z)$ or $F(x, u)$ (resp. $W(x, u)$) only manages to solve to optimality 18 (resp. 24) of them. Among the 80 instances with $n = 20$, formulation $LF(x, u)$ solves 57 instances to optimality, while $F(x, u, z)$ (resp. $W(x, u)$, $F(x, u)$) only manage to solve to optimality 49 (resp. 42, 34) of them. Formulation $\text{Int}(\tilde{y})$ solves to optimality only 20 of them. Formulation $LF(x, u)$ also globally improves the solving time. For example, on instances of size $(n, T) = (60, 48)$ and $F = 3$, the average CPU time of formulation $LF(x, u)$ is 1450 seconds, while this number increases to 3422 (resp. 2527, 2689) for $F(x, u)$ (resp. $F(x, u, z)$, $W(x, u)$).

As there is an important variability in the computation time for instances with same size (n, T) and same F , we introduce the improvement score. For given formulations F_1 and F_2 , the *improvement score* I of F_1 with respect to F_2 is as follows.

$$I = 2 \frac{\text{CPU}(F_2) - \text{CPU}(F_1)}{\text{CPU}(F_2) + \text{CPU}(F_1)}$$

The improvement score I is a performance ratio comparing formulation CPU times pairwise.

Table 6 provides, for each formulation $F \in \{ F(x, u, z), LF(x, u) \}$, the average improvement score of F with respect to $F(x, u)$ on each group of 20 instances. Formulation $\text{Int}(\tilde{y})$ is not included as on most instance groups, it solves no instance to optimality.

Formulation $LF(x, u)$ outperforms all other formulations. In particular, even if on $(20, 96)$ and $F = 2$ instances the average CPU time of $LF(x, u)$ is slightly higher than $F(x, u, z)$, the average improvement score of $LF(x, u)$ is more important. This shows that $LF(x, u)$ have higher CPU time than $F(x, u, z)$ on instances on which the difference in CPU time is not very significant with respect to Cplex's CPU time. On the opposite, $LF(x, u)$ have low CPU time on instances on which this difference in CPU time represents an important improvement. Note that formulation $W(x, u)$ appears to perform better than $F(x, u, z)$ on $T = 48$ instances. Recall that $W(x, u)$ is only partial-symmetry-breaking. Thus, when T is smaller, the number of feasible columns featuring a given number of 1-entries is also smaller. On the opposite, when $T = 96$, the number of one-entries is not a very discriminating indicator among symmetric columns. Therefore $W(x, u)$ is not able to break as much symmetries, and $F(x, u, z)$ globally performs better.

For example, on $(n, T) = (60, 48)$, $F = 4$ instances, the improvement score of $LF(x, u)$ is 109%, while it is 45.8% for $F(x, u, z)$ and 27.9% for $W(x, u)$. On $(n, T) = (30, 48)$, $F = 3$ instances, the improvement score of $LF(x, u)$ is 114%, while it is -19.3% for $F(x, u, z)$ and 25.3% for $W(x, u)$. On $(n, T) = (20, 96)$, $F = 3$ instances, this number increases to 47.6% for $LF(x, u)$ (resp. 22.9% for $F(x, u, z)$, -11.9% for $W(x, u)$).

Instances		Formulation	#opt	#nodes	gap (%)	CPU time
(20,48)	$F = 2$	$F(x, u)$	9	667 974	0.009 16	2061.6
		$\text{Int}(\tilde{y})$	9	22 583	0.049 10	2426.1
		$W(x, u)$	10	232 589	0.011 15	1965.2
		$F(x, u, z)$	11	139 493	0.009 91	1840.4
		$LF(x, u)$	16	242 096	0.001 89	980.4
	$F = 3$	$F(x, u)$	13	634 436	0.002 96	1424.7
		$\text{Int}(\tilde{y})$	3	7239	5.072 07	3243.3
		$W(x, u)$	16	314 447	0.004 40	1295.9
		$F(x, u, z)$	18	102 717	0.002 26	998.0
		$LF(x, u)$	20	30 014	0	132.8
(20,96)	$F = 2$	$F(x, u)$	5	702 415	0.007 76	2781.9
		$\text{Int}(\tilde{y})$	3	10 148	0.027 54	3188.6
		$W(x, u)$	4	233 582	0.025 84	3058.1
		$F(x, u, z)$	8	61 384	0.006 81	2556.5
		$LF(x, u)$	6	160 150	0.007 18	2675.6
	$F = 3$	$F(x, u)$	7	989 738	0.006 44	2470.2
		$\text{Int}(\tilde{y})$	5	16 776	10.067 68	3109.4
		$W(x, u)$	5	198 137	0.014 66	2725.6
		$F(x, u, z)$	12	87 375	0.004 24	1819.7
		$LF(x, u)$	15	186 018	0.005 65	1794.7
(30,48)	$F = 2$	$F(x, u)$	4	354 029	0.018 03	2924.7
		$\text{Int}(\tilde{y})$	0	0	100	-
		$W(x, u)$	7	210 032	0.011 00	2535.4
		$F(x, u, z)$	4	71 467	0.025 47	2969.0
		$LF(x, u)$	15	219 655	0.002 04	1341.8
	$F = 3$	$F(x, u)$	6	379 482	0.012 13	2676.9
		$\text{Int}(\tilde{y})$	0	0	100	-
		$W(x, u)$	10	240 767	0.006 98	1931.4
		$F(x, u, z)$	5	107 609	0.016 23	2736.1
		$LF(x, u)$	16	191 113	0.002 19	965.7
(30,96)	$F = 2$	$F(x, u)$	3	390 666	0.004 63	3069.8
		$\text{Int}(\tilde{y})$	0	0	100	-
		$W(x, u)$	4	121 205	0.007 55	3130.1
		$F(x, u, z)$	5	46 869	0.009 18	3107.7
		$LF(x, u)$	9	315 503	0.002 38	2263.5
	$F = 3$	$F(x, u)$	5	460 304	0.003 24	2927.0
		$\text{Int}(\tilde{y})$	0	0	100	-
		$W(x, u)$	3	211 303	0.004 65	3130.5
		$F(x, u, z)$	4	61 994	0.004 55	3059.7
		$LF(x, u)$	12	183 633	0.000 77	1852.9

Table 4: Performance indicators relative to the comparison of five formulations for ramp-constrained MUCP instances with symmetries and $n = 20, 30$

Instances		Formulation	#opt	#nodes	gap (%)	CPU time
(60,48)	$F = 2$	$F(x, u)$	1	757 017	0.003 09	3437.6
		$\text{Int}(\tilde{y})$	0	7919	0.030 78	-
		$W(x, u)$	4	203 485	0.002 85	3046.2
		$F(x, u, z)$	6	66 272	0.037 46	2839.8
		$LF(x, u)$	5	569 546	0.001 26	2710.6
	$F = 3$	$F(x, u)$	1	850 192	0.002 68	3422.5
		$\text{Int}(\tilde{y})$	1	8300	5.181 95	3523.9
		$W(x, u)$	6	192 656	0.002 45	2689.3
		$F(x, u, z)$	9	40 680	0.003 97	2527.5
		$LF(x, u)$	14	493 254	0.000 40	1450.2
	$F = 4$	$F(x, u)$	7	870 666	0.002 43	2582.4
		$\text{Int}(\tilde{y})$	0	1236	25.951 57	-
		$W(x, u)$	10	295 149	0.000 95	1971.9
		$F(x, u, z)$	14	33 574	0.000 53	1623.1
		$LF(x, u)$	15	459 142	0.000 27	1043.8
(60,96)	$F = 2$	$F(x, u)$	0	120 125	0.012 62	-
		$\text{Int}(\tilde{y})$	0	0	100	-
		$W(x, u)$	0	23 851	0.051 90	-
		$F(x, u, z)$	0	3813	0.528 55	-
		$LF(x, u)$	0	52 226	0.011 25	-
	$F = 3$	$F(x, u)$	0	144 265	0.014 90	-
		$\text{Int}(\tilde{y})$	0	0	100	-
		$W(x, u)$	0	50 841	0.018 15	-
		$F(x, u, z)$	0	6404	0.034 76	-
		$LF(x, u)$	0	83 335	0.013 11	-
	$F = 4$	$F(x, u)$	0	230 935	0.009 56	-
		$\text{Int}(\tilde{y})$	0	0	100	-
		$W(x, u)$	0	92 298	0.010 63	-
		$F(x, u, z)$	0	9616	0.015 89	-
		$LF(x, u)$	2	150 692	0.006 56	3467.7

Table 5: Performance indicators relative to the comparison of five formulations for ramp-constrained MUCP instances with symmetries and $n = 60$

Instances		Improvement score w.r.t. $F(x, u)$		
		$W(x, u)$	$F(x, u, z)$	$LF(x, u)$
(20,48)	$F = 2$	-6.61%	-14.2%	83.7%
	$F = 3$	-11.3%	-12.2%	111%
(20,96)	$F = 2$	-13%	9.05%	23.9%
	$F = 3$	-11.3%	22.9%	47.6%
(30,48)	$F = 2$	20.8%	-17.8%	89.4%
	$F = 3$	25.3%	-19.3%	114%
(30,96)	$F = 2$	-15.8%	-19.4%	40.4%
	$F = 3$	-11.8%	-7.63%	76.4%
(60,48)	$F = 2$	26.5%	26.2%	47.8%
	$F = 3$	35.5%	30.9%	104%
	$F = 4$	27.9%	45.8%	109%
(60,96)	$F = 2$	0%	0%	0%
	$F = 3$	0%	0%	0%
	$F = 4$	0%	0%	4.72%

Table 6: Improvement scores of formulations $W(x, u)$, $F(x, u, z)$ and $LF(x, u)$ w.r.t formulation $F(x, u)$ for ramp-constrained MUCP instances with symmetries

6 Conclusion

We propose a framework to build sub-symmetry-breaking inequalities, in order to handle the symmetries arising from a collection of sub-symmetric solution subsets. These inequalities may require to introduce one additional variable z per solution subset considered. Depending on the subset structure, variable z may only be a linear expression of variables x , and therefore does not need to be added to the model as an additional variable. The derived sub-symmetry-breaking inequalities are full symmetry-breaking under a mild condition. If this condition is not satisfied anyway, a new collection of sub-symmetric subsets can be constructed such that the derived inequalities are full symmetry-breaking.

Our experimental results for the MUCP show that aggregation of the classical formulation is a very efficient technique to handle symmetries and sub-symmetries arising in the MUCP. When ramp constraints are taken into account in the MUCP, disaggregation is no longer possible. Our sub-symmetry-breaking inequalities can still be used and outperform all other formulations.

Sub-symmetry-breaking inequalities are always applicable as the solution subsets considered can capture the specific conditions under which the symmetries hold. On the opposite, aggregated formulations require specific conditions to be applicable.

One perspective is to use the framework proposed in this article to derive new sub-symmetry-breaking inequalities for “ready to shut down” sub-symmetries in the ramp-constrained case. Another perspective is to apply the proposed framework to other problems featuring sub-symmetric solution subsets such as covering problems, or bin packing variants where one item can be placed in multiple bins. It would also be useful to study how the presented framework could be automated, so that sub-symmetric subsets are automatically detected and variables z automatically constructed.

References

- [1] S. Baum and L. E. Trotter. Integer rounding and polyhedral decomposition for totally unimodular systems. In *Optimization and Operations Research*, pages 15–23. Springer, 1978.
- [2] P. Bendotti, P. Fouilhoux, and C. Rottner. The min-up/min-down unit commitment polytope. *Journal of Combinatorial Optimization*, 36(3):1024–1058, 2018.
- [3] P. Bendotti, P. Fouilhoux, and C. Rottner. On the complexity of the unit commitment problem. *Annals of Operations Research*, 2018.
- [4] P. Bendotti, P. Fouilhoux, and C. Rottner. Orbitopal fixing for the full (sub-)orbitope and application to the unit commitment problem. *Optimization Online*, 2018.
- [5] T. Berthold and M. E. Pfetsch. Detecting orbitopal symmetries. In *Operations Research Proceedings 2008*, pages 433–438. Springer, 2009.
- [6] M. Carrion and J. M. Arroyo. A computationally efficient mixed-integer linear formulation for the thermal unit commitment problem. *IEEE Transactions on Power Systems*, 21, 2006.
- [7] M. Fischetti, L. Liberti, D. Salvagnin, and T. Walsh. Orbital shrinking: Theory and applications. *Discrete Applied Mathematics*, 222:109 – 123, 2017.
- [8] E. J. Friedman. *Fundamental Domains for Integer Programs with Symmetries*, pages 146–153. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.
- [9] C. Hojny and M. E. Pfetsch. Polytopes associated with symmetry handling. *Optimization Online*, 2017. http://www.optimization-online.org/DB_HTML/2017/01/5835.html.
- [10] Raf Jans. Solving lot-sizing problems on parallel identical machines using symmetry-breaking constraints. *INFORMS Journal on Computing*, 21(1):123–136, 2009.
- [11] V. Kaibel and A. Loos. Branched polyhedral systems. In *Proceedings of the 14th International IPCO Conference on Integer Programming and Combinatorial Optimization*. Springer-Verlag, 2010.
- [12] V. Kaibel, M. Peinhardt, and M. E. Pfetsch. Orbitopal fixing. In *IPCO*, pages 74–88. Springer, 2007.
- [13] V. Kaibel and M. E. Pfetsch. Packing and partitioning orbitopes. *Mathematical Programming*, 114(1):1 – 36, 2008.
- [14] B. Knueven, J. Ostrowski, and J. Wang. Generating cuts from the ramping polytope for the unit commitment problem. *Optimization Online*, 2016. http://www.optimization-online.org/DB_HTML/2015/09/5099.html.
- [15] B. Knueven, J. Ostrowski, and J. P. Watson. Exploiting identical generators in unit commitment. *IEEE Transactions on Power Systems*, pages 1–1, 2017.
- [16] L. Liberti. Reformulations in mathematical programming: automatic symmetry detection and exploitation. *Mathematical Programming*, 131(1):273–304, 2012.

- [17] L. Liberti and J. Ostrowski. Stabilizer-based symmetry breaking constraints for mathematical programs. *Journal of Global Optimization*, 60(2):183–194, 2014.
- [18] R. M. Lima and A. Q. Novais. Symmetry breaking in MILP formulations for Unit Commitment problems. *Computers & Chemical Engineering*, 85:162–176, 2016.
- [19] A. Loos. *Describing Orbitopes by Linear Inequalities and Projection Based Tools*. PhD thesis, Universität Magdeburg, 2011.
- [20] F. Margot. Exploiting orbits in symmetric ILP. *Mathematical Programming*, 98(1):3–21, 2003.
- [21] J. Ostrowski, M. F. Anjos, and A. Vannelli. Tight mixed integer linear programming formulations for the unit commitment problem. *IEEE Transactions on Power Systems*, 27, 2012.
- [22] J. Ostrowski, M.F. Anjos, and A. Vannelli. Modified orbital branching for structured symmetry with an application to unit commitment. *Mathematical Programming*, 150(1):99 – 129, 2015.
- [23] J. Ostrowski, J. Linderoth, F. Rossi, and S. Smriglio. Orbital branching. *Mathematical Programming*, 126(1):147–178, 2011.
- [24] K. Pan and Y. Guan. A polyhedral study of the integrated minimum-up/-down time and ramping polytope. *Optimization Online*, 2016. http://www.optimization-online.org/DB_HTML/2015/08/5070.html.
- [25] D. Rajan and S. Takriti. Minimum up/down polytopes of the unit commitment problem with start-up costs. *IBM Research Report*, 2005.