Arities of Symmetry Breaking Constraints

Tim Januschowski*

Cork Constraint Computation Centre
Computer Science Department
University College Cork Ireland
janus@cs.ucc.ie

Abstract. Static symmetry breaking is a well-established technique to speed up the solving process of symmetric Constraint Satisfaction Programs (CSPs). Static symmetry breaking suffers from two inherent problems: symmetry breaking constraints come in great numbers and are of high arity. Here, we consider the problem of high arity. We prove that not even for binary CSPs can we always reduce the arity of the commonly used lexleader constraints. We further prove that for binary CSPs we sometimes have to rely on at least ternary constraints to break all symmetries. On the positive side, we prove that symmetry breaking constraints with arity $\lfloor n/2 \rfloor + 1$ exist that always break the symmetries of a CSP completely. For various special cases of CSPs, we prove that binary symmetry breaking constraints may break all symmetries.

1 Introduction

Symmetry breaking has been the subject of intense investigation for almost two decades in Constraint Programming, see e.g. [5]. Symmetry breaking has an empirically proved potential to speed up constructive search methods. The classic and practically most used technique in symmetry breaking is the addition of symmetry breaking constraints before search [2, 11].

For complete symmetry breaking, we have to add one lexleader constraint (LLC) [2] per symmetry, and the arity of each symmetry breaking constraint is the number of variables. In the worst case, a CSP can have an exponential number of symmetries. Adding an exponential number of constraints to a CSP is prohibitively costly. High arity slows down propagation [3]. Various remedies have been proposed: for special symmetry groups, we can find polynomial sized sets of constraints with reasonable arity that break all symmetries [6], or sometimes one can use the problem structure in combination with the symmetries to reduce the number and arity of the constraints [8, 12]. Also, various reduction rules [4, 6, 10] are known that reduce both the number and the arity of LLCs while maintaining the ability to completely break symmetries. Another remedy that we do not want to consider here is partial symmetry breaking.

---

* Tim Januschowski is supported by the Embark initiative of the Irish Research Council for Science, Engineering and Technology.

In this paper, we consider upper and lower bounds on the arities of symmetry breaking constraints for complete symmetry breaking in binary CSPs. Our lower bounds hold for *any* symmetry breaking constraint and *any* reduction rule. We prove the following for CSPs with n variables:

upper bound (i): the arity of LLCs cannot always be reduced, even if we assume the CSP to be binary,

upper bound (ii): symmetry breaking constraints of arity $\lfloor n/2 \rfloor + 1$ exist that completely break the symmetries of any CSP,

lower bound: for complete symmetry breaking in binary CSPs, we sometimes have to rely on ternary constraints, and

special cases: we identify various special cases, where binary symmetry breaking constraints suffice for complete symmetry breaking.

To the best of our knowledge, these results are new. We think that our results are an important theoretical contribution to the study of symmetry breaking constraints. Our results could be applied to study the optimality of reduction rules in terms of the arity of symmetry breaking constraints. If a reduction rule can reduce the arity of symmetry breaking constraints for all binary CSPs to at most three, the reduction rule could be considered optimal. On the other hand, if the reduction rule does not succeed to reduce the arity of symmetry breaking constraints below $\lfloor n/2 \rfloor + 1$ for all CSPs, then the rule is surely not optimal.

2 Notation and Definitions

A *constraint satisfaction problem* CSP is a triple $(V, D, Cons)$, where V is the set of variables of the CSP, every variable x has a *domain* $D(x) \in D$, and *Cons* is the set of constraint of the CSP. Every constraint has an *arity*. The k-ary constraint c is a pair $\langle s, r \rangle$, where s is a list of k variables $x_1, \ldots, x_k$ which is called the *scope* and $r \subseteq D(x_1) \times \cdots \times D(x_k)$ is called *the relation* of c consisting of the tuples that c allows. The arity of a CSP is the maximum arity over all constraints in the CSP. A CSP is called *binary* if all constraints are of arity at most 2. A *literal* is a (variable,value)-assignment. A *partial assignment* is a set of literals in which no variable appears twice. If a partial assignment is allowed by the constraints of the CSP we call it *consistent*. A *solution* is a consistent assignment on all variables. If a CSP has a solution, the CSP is *satisfiable*, otherwise it is *unsatisfiable*.

We associate to any CSP a hypergraph called the *microstructure complement* (MSC), see e.g. [1]. The MSC has as nodes the literals of the CSP. We have a hyperedge between every set of literals that is not contained in the relation of a constraint of the CSP. For a binary CSP, the MSC is a graph and for binary CSPs we define the *microstructure* as the complement graph of the MSC. The *constraint symmetries* [1] of a CSP are the automorphisms of the MSC. Symmetries partition the set of solutions of a CSP into a set of equivalences classes or *orbits*. For a solution T, we denote by $\text{orbit}(T)$ the set of solutions that are symmetric to T. Apart from constraint symmetries, other symmetries exist as well, notably solution symmetries. However, constraint programmers work with constraint symmetries mostly [1]. Here, we only consider constraint symmetries which we shall abbreviate to symmetries.

Given a CSP P, a *valid reduction* P' [7, 11] is a CSP on the same variables, subsets of the domains and supersets of the constraints of P, such that for every orbit of solutions in P, at least one solution in P' exists. Among the solutions in P' that exist for every orbit of solutions in P, we can choose one as an *orbit-representative solution* per orbit in P. A *single-representative valid reduction* (SRVR) is a valid reduction that has *exactly* one solution per orbit of solutions in P. If P' is a SRVR, the set of solutions of P' is a *complete* set of orbit representative solutions.

We call members of a family of constraints *symmetry breaking constraints*, if adding the family to a CSP leads to a valid reduction. Lexleader constraints (LLCs) [2, 13] are well-known symmetry breaking constraints. To construct a LLC for a symmetry ϕ of a CSP, we choose a variable order, say $x_1 \prec x_2 \prec \cdots \prec x_n$ and an order on the domains. The LLC enforces that any assignment on the variables is lexicographically less than its symmetric counterpart with respect to the chosen orders:

$$([x_1, \ldots, x_n]) \leq_{lex} \phi([x_1, \ldots, x_n]) \, .$$

We note that ϕ may not map a complete assignment to another complete assignment. In this case however, the assignment under consideration cannot be a solution and must be forbidden by some constraints of the CSP under consideration, otherwise ϕ would not be a symmetry.

SRVRs exist for any CSP and we can always find a SRVR using LLCs. With LLCs, the choice of orbit representative solutions depends on the order of the variables that we choose to define the LLCs as well as the orders on the domains. However, also for arbitrary choices of orbit representatives, a SRVR always exists: we could simply add an n-ary constraint forbidding any non-representative solution [7]. If the arity of the constraints we add to a CSP to obtain the SRVR is at most k, we say that the SRVR is *k-ary realisable*.

In the context of this paper, we consider the *full* group of (constraint) symmetries. Our intuition is that the full group of symmetries is more closely connected to the MSC than *sub*groups. Results obtained for subgroups may not hold for the full group as orbits of solutions tend to interact in a less complicated way for smaller groups. Indeed, the number of orbits of solutions typically decreases for increasing sizes of subgroups. This also helps to explain, why it seems more difficult to obtain results for the full group of symmetries than for subgroups. In this paper, we investigate *complete* symmetry breaking, or rather the existence of SRVRs.

3 Related Work

This work is based on Puget's systematic approach to introduce symmetry breaking constraints via valid reductions [11]. This approach was generalised and extended in [7]. There, we also posed the question, whether for any k-ary CSP, there always is a k-ary realisable SRVR. Here, we will give a negative answer to

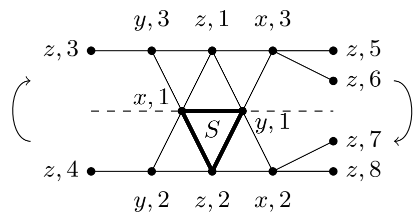


Fig. 1: A CSP with 3 variables where we have to use a ternary LLC in order to disallow the thickly edged solution S, see Example 1. The only symmetry that maps solutions to solutions is a reflection about the dashed line.

this question for the case $k = 2$. However, we also provide a range of special cases where we can answer the question affirmatively.

Reduction rules for LLCs have been studied in [4, 6, 10]. These reduction rules reduce the arity and number of LLCs. Grayland *et al.* [6] manage to show the minimality of certain sets of constraints with respect to the reduction rules under consideration. In a way, we study minimality of symmetry breaking constraints both independently of concrete reduction rules and independently of concrete constraints used for symmetry breaking. Solely based on the way orbits of solutions may interact, we show that LLCs exist whose arity cannot be reduced.

We consider the arity of symmetry breaking constraints on the original variables. For value symmetries, the introduction of auxiliary variables has been shown to be useful for finding sets of sometimes binary symmetry breaking constraints [9]. However, as we are interested in general results here, we focus on the original variables: any constraint can be turned into a binary constraint with the help of auxiliary variables.

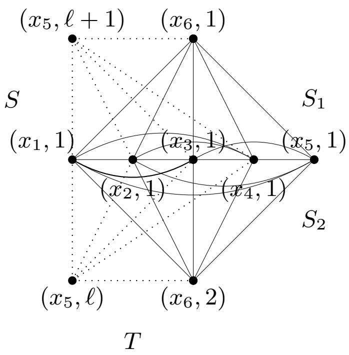
4 Upper Bounds on the Arities

In this section, we want to consider upper bounds on the arities of symmetry breaking constraints. More precisely, we prove that for the commonly used LLCs with a given variable order, no reduction rule may always reduce the arity. We show, however, that symmetry breaking constraints always exist whose arity does not exceed $\lfloor n/2 \rfloor + 1$.

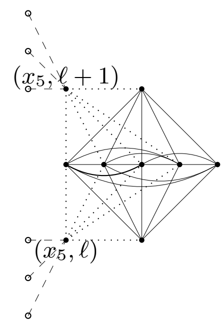
4.1 Fixed-order LLCs Are Intrinsically n-ary

In this section, we show that we cannot reduce the arity of LLCs in all CSPs. Before giving a general proof, let us consider the following example.

Example 1. Let $P = (\{x, y, z\}, D, Cons)$. We have $D(x) = D(y) = \{1, 2, 3\}$, and $D(z) = \{1, 2, \ldots, 8\}$. The constraints are such that P has a microstructure as depicted in Figure 1. By inspection, it is obvious that the only symmetry that



(a) The initial solutions S_1 and S_2, with thin edges, contain all literals corresponding to assignments of value 1 to all variables in the set $\{x_1, \ldots, x_5\}$ and value 1, respectively 2 to variable x_6. Here, we also depict two solutions S and T with dotted edges. Solution S contains all literals of S_1 except for $(x_5, 1)$ and solution T is the symmetric counterpart of S under the reflection.



(b) To ensure that the solutions S and T are only symmetric to themselves, we add more literals to the literals (x_5, ℓ), and $(x_5, \ell+1)$, we depict the edges as dashed.

Fig. 2: Construction of the CSP as in the proof of Theorem 1 for the 6 variable case. The only symmetry that maps solutions to different solutions is a reflection about an axis going through $\{(x_1, 1), \ldots, (x_5, 1)\}$.

permutes a solution to another solution is a reflection about the dashed line. We choose $x \prec y \prec z$ and the natural order on the integers for lexicographic comparison. The solution $S = \{(x, 1), (y, 1), (z, 2)\}$ is not a lex-minimal solution in its orbit, but all pairs of literals in S are contained in lex-minimal solutions. In order to exclude any solution that is not lex-minimal, as LLCs would, we need a ternary constraint in order to disallow S.

We generalise the example in the following and show that for every $n > 1$, we can find a CSP with the same property that the CSP in Example 1 has: we need n-ary constraints to obtain a SRVR whose orbit representative solutions are the lex-minimal ones. We illustrate our proof with a binary CSP with 6 variables whose microstructure we partially depict in Figure 2. From the construction of the CSP in our proof, it will be clear that we cannot expect LLCs for a *binary* CSP with n variables to have arity less than n. This means that the various reduction rules that are known to reduce the complexity of the LLCs will not be able to reduce the arity of LLCs for all CSPs—in fact, we show that this is true for *any possible* reduction rule.

Theorem 1. *For any $n > 1$, a binary CSP with n variables and an order on its variables exist such that any SRVR that has lex-minimal (wrt to the variable order) solutions as orbit representative solutions is n-ary realisable.*

Proof. We construct a CSP with n variables ordered as $x_1 \prec x_2 \prec \cdots \prec x_n$. We construct the domains of the variables during the proof and we use the natural order on the integers for lexicographic comparison. Whenever we say that we construct a solution of the CSP, we really mean that we add or modify constraints to the CSP such that they allow these solutions. We construct the solutions of the CSP in such a way that the symmetries of the CSP will be immediate. We start with two solutions $S_i = \{(x_1, 1), \ldots, (x_{n-1}, 1), (x_n, i)\}$ where $i = 1, 2$. We have depicted the solutions S_1 and S_2 in Figure 2a for the case where $n = 6$.

Our aim is to add solutions to the CSP such that every subset of size up to $n - 1$ in S_2 is contained in a lex-minimal solution. When we add the solutions, care has to be taken in two aspects. First, we need to ensure that S_1 and S_2 remain symmetric under the symmetry of the CSP that swaps $(x_n, 1)$ with $(x_n, 2)$ while leaving all other literals unchanged. We call this symmetry a *reflection* due to the geometric intuition, see Figure 2. Every time we add literals to the CSP, we have to ensure that we can extend the reflection to the new literals. Second, we need to ensure that S_1 and S_2 are symmetric only to each other. Solution S_1 is, with respect to the chosen orders, lex-less than S_2. Hence, we need to exclude S_2. Having constructed a lex-minimal solution for every subset of size up to $n - 1$ in S_2, it follows that if we want to find a SRVR with the lex-minimal orbit representatives, we have to use an n-ary constraint to remove S_2.

We construct symmetric solutions for symmetric subsets of solutions S_1 and S_2 with size up to $n - 1$. We first note that any subset of S_2 not involving literal $(x_n, 2)$ is contained in S_1 by construction and hence, we cannot add a constraint that forbids this subset. We note next that it suffices to consider only subsets of size exactly $n - 1$. For any subset of size $n - 1$ that includes literal $(x_n, 2)$, we construct a solution that uses this subset. For the subset of size $n - 1$ to be a solution, we introduce a literal (x, ℓ), such that ℓ does not yet appear in the CSP and ensure that the subset with the new literal is a solution T. We construct another solution S containing all literals corresponding to value 1 in T, it contains $(x_n, 1)$ as well as a new literal $(x, \ell + 1)$. We ensure that S is symmetric to T with respect to the reflection by extending the reflection to swap (x, ℓ) and $(x, \ell+1)$. In Figure 2a, we have depicted T for the subset of 5 literals of solution S_2 consisting of the set of literals $\{(x_1, 1), (x_2, 1), (x_3, 1), (x_4, 1), (x_6, 2)\}$. Solution T contains the new literal (x_5, ℓ), where ℓ is chosen appropriately.

With our construction so far, we have ensured that T and S are symmetric and that S_1 and S_2 remain symmetric. To ensure that T and S are not symmetric to other solutions, we add a set of literals L. In the microstructure, we make half of the literals in L adjacent to (x, ℓ) and the other half adjacent to $(x, \ell + 1)$. We choose the cardinality of L in such a way that (x, ℓ) and $(x, \ell + 1)$ have a (node) degree in the microstructure that only these two literals have, see Figure 2b. By addition of the literals, the CSP gains more non-trivial symmetries that permute the recently added literals while leaving all other literals unchanged. However, we do not have to consider these new symmetries, as they are the identity when restricted to solutions.

With this construction, solutions S_1 and S_2 remain symmetric. Furthermore, solutions S and T are the only members of their orbit, due to the degrees of the literals (x, ℓ), and $(x, \ell+1)$. The lex-minimal solution in this orbit is T. Hence, we cannot forbid any set of literals of size up to $n-1$ in S_2. This shows that the constraint that forbids solution S_2 must be n-ary. □

In other words, Theorem 1 shows that no reduction rule will be able to reduce the arity of LLCs below the original arity for all CSPs. The step of the proof depicted in Figure 2b is not necessary, if we want to prove the theorem only for a subgroup of the symmetry group.

4.2 A $\lfloor n/2 \rfloor + 1$-ary Realisable SRVR Always Exist

In this section, we prove the existence of symmetry breaking constraints with arity $\lfloor n/2 \rfloor + 1$. In order to show this, we need more notation.

In the following we introduce distinguished sets of orbit representative solutions for a CSP. We refer to the set of literals of a CSP by *lits*. For a complete set of orbit representative solutions R, we define a function $c(\cdot, R)$. For a literal $v \in lit$, we have $c(v, R) = 0$ if v is contained in at most one solution in R. If v is contained in k orbit representative solutions in R, where $k \geq 2$, then $c(v, R) = k-1$. We define $c(R) = \sum_{v \in lits} c(v, R)$. For the set $\mathcal{S}$ of complete sets of orbit representative solutions, we call a set $R \in \mathcal{S}$ such that $R = \arg\max_{Q \in \mathcal{S}} c(Q)$ a *max-common set*. The advantage of a max-common set is that it helps to break down the globality of the symmetries into some desirable "local" properties which we will then explore. We need the following observation to prove such a property.

Proposition 1. *Let S and T be two solutions and let ϕ be any symmetry of the CSP. Then $|S \cap T| = |\phi(S) \cap \phi(T)|$.*

This is enough to prove following property of any max-common set.

Proposition 2. *For a max-common set of orbit representative solutions $R = \{S_1, \ldots, S_t\}$ and a non-representative solution $T \in \text{orbit}(S_j)$, any S_i with $|S_i \cap T| = k$ has $|S_i \cap S_j| \geq k$.*

Proof. By contradiction. Let S_i be a solution in a max-common set such that $|S_i \cap T| = k$. We assume that $|S_i \cap S_j| < k$. Let ϕ be a symmetry such that $\phi(S_j) = T$. By Proposition 1, we know that $|\phi(S_i) \cap \phi(S_j)| = k$. Hence, for the set $\phi(R) := \{\phi(S_1), \ldots, \phi(S_t)\}$, we have $c(\phi(R)) > c(R)$. This is a contradiction to the fact that $S_1, \ldots, S_t$ is a max-common set of orbit representative solutions.

We can now prove the main result of this section.

Theorem 2. *A $\lfloor n/2 \rfloor + 1$-ary realisable SRVR exists for any CSP with n variables.*

Proof. Consider a CSP with n variables and a max-common set of orbit representative solutions R. Let T be any solution with $T \notin R$. Let $S \in R$ be such that $T \in \text{orbit}(S)$..

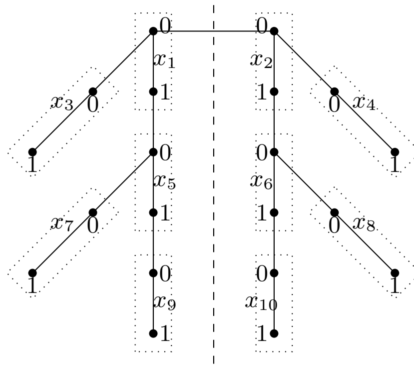


Fig. 3: The MSC of a binary CSP for which no binary realisable SRVR exists. The only symmetry of the CSP is a reflection about the dashed line. Nodes in the dotted boxes are literals corresponding to the same variable.

Let n be even. We assume first that $|S \cap T| < n/2$. We show that a subset of $n/2 + 1$ literals in $T \setminus S$ exists that is not contained in any orbit representative in R. For the sake of contradiction, we assume that no such subset exists, so we assume all $n/2 + 1$ literals are contained in a member of R. By Proposition 2, any $S_1 \in R$ that contains $n/2+1$ literals in $T \setminus S$ contains at least $n/2+1$ literals in $S \setminus T$. Hence, S_1 contains more than n literals, which is a contradiction. This means that a set of literals in T exists that is not in any orbit representative solution and we can forbid this set safely.

Next, we assume that $|S \cap T| \geq n/2$. We show that all of the, say k, literals in $T \setminus S$ and $n/2 - k$ literals in $T \cap S$ cannot be contained in any $S_1 \in R$. If the $n/2$ literals were contained in S_1, then S_1 would also contain at least k literals in $S \setminus T$. However, the literals in $S \setminus T$ have the same variables as the k literals in $T \setminus S$. Therefore, S_1 is not a solution, which is a contradiction and we can forbid a set of $n/2 + 1$ literals in T safely.

For an odd number of variables the argumentation is similar and we omit this case due to space restrictions. □

5 A Lower Bound on the Arity

In this section, we show that for binary CSPs we sometimes have to rely on non-binary constraints to break all symmetries. This provides a lower bound for the afore-mentioned reduction rules: If a reduction rule for binary CSPs can reduce the arity of symmetry breaking constraints for any CSP to the lower bound we provide here, then the reduction rule is optimal.

Theorem 3. *A binary CSP exists that does not have a binary realisable SRVR.*

Proof. We consider a 10 variable CSP P_{tree} where every variable has domains $\{0, 1\}$. The MSC of P_{tree} is a tree. We depict it in Figure 3. By inspection of the MSC, it is obvious that there is only one non-trivial symmetry of the MSC. It swaps literals (x_i, j) with (x_{i+1}, j) for odd i and $j \in \{0, 1\}$. The symmetry is a reflection about the dotted line in Figure 3. We define a set $L := \cup_{i \in \{1,2,5,6,9,10\}} \{(x_i, 1)\}$. The solutions of P_{tree} we consider in the following, have L in common. We consider the following four self-symmetric solutions of P_{tree}:

$$L \cup \{(x_3, 1), (x_4, 1), (x_7, 0), (x_8, 0)\}\,, \qquad L \cup \{(x_3, 1), (x_4, 1), (x_7, 1), (x_8, 1)\}\,,$$
$$L \cup \{(x_3, 0), (x_4, 0), (x_7, 1), (x_8, 1)\}\,, \text{ and } \quad L \cup \{(x_3, 0), (x_4, 0), (x_7, 0), (x_8, 0)\}.$$

Orbits of solutions that consist of more than one solution contain exactly two solutions because the CSP only has one non-trivial symmetry. From the orbits of solutions with two members, we consider the following eight solutions:

$$\begin{aligned}
A_1 &= L \cup \{(x_3, 1), (x_4, 0), (x_7, 1), (x_8, 1)\}\,, \\
A_2 &= L \cup \{(x_3, 0), (x_4, 1), (x_7, 1), (x_8, 1)\}, \\
B_1 &= L \cup \{(x_3, 1), (x_4, 1), (x_7, 0), (x_8, 1)\}\,, \\
B_2 &= L \cup \{(x_3, 1), (x_4, 1), (x_7, 1), (x_8, 0)\}, \\
C_1 &= L \cup \{(x_3, 1), (x_4, 0), (x_7, 0), (x_8, 1)\}\,, \\
C_2 &= L \cup \{(x_3, 0), (x_4, 1), (x_7, 1), (x_8, 0)\}, \\
D_1 &= L \cup \{(x_3, 1), (x_4, 0), (x_7, 1), (x_8, 0)\}\,, \\
D_2 &= L \cup \{(x_3, 0), (x_4, 1), (x_7, 0), (x_8, 1)\}\,.
\end{aligned}$$

Members of the same orbit have the same capital letter. In order to find a SRVR we need to choose an orbit representative among solutions A_1 and A_2. The only pair of literals in A_1 that is not contained in any of the four self-symmetric solutions is pair $p := \{(x_3, 1), (x_4, 0)\}$. Disallowing p, makes A_2 the orbit representative of $\{A_1, A_2\}$ and removes C_1 and D_1 and hence, C_2 and D_2 must be the orbit representatives of the respective orbits. Consider $\{B_1, B_2\}$ next. The only pair of literals in B_1 that we can forbid is the pair of literals $q := \{(x_7, 0), (x_8, 1)\}$. Disallowing q however would disallow solution D_2, an orbit representative. The same is true for B_2. The only pair of literals we could disallow is $\{(x_7, 1), (x_8, 0)\}$ which would lead to orbit representative C_2 being forbidden. Hence, solution A_2 cannot be an orbit representative.

A symmetric argumentation holds if we choose solution A_2 as an orbit representative, which shows that P_{tree} does not have a binary realisable SRVR. □

The CSP P_{tree} in the proof of Theorem 3 has rather many solutions and orbits. Unfortunately, our attempts to construct a less complex example have failed. In the next section, we shall see reasons for the complexity of the example: for various special cases of CSPs, binary realisable SRVRs always exist.

6 Special Cases

In this section, we present special cases in which binary realisable SRVRs exist.

Theorem 4 (Sufficient Conditions). *In all of the following cases, a (not necessarily binary)* CSP P *has a binary realisable* SRVR*:*

- *for any pair of solutions S, T of P from different orbits we have $S \cap T = \emptyset$,*
- *P has only one orbit of solutions with cardinality greater or equal to 1, and*
- *P has exactly 2 orbits of solutions.*

Proof. If solutions in different orbits do not share literals, we can consider each orbit on its own. In each orbit, we choose an orbit representative S. Any other solution T in orbit(S) differs from S in at least one literal ℓ. Since solutions from different orbits are disjoint from T, literal ℓ is not contained in any orbit representative. Hence, we can forbid this literal and even unary constraints suffice.

The second case is a corollary of Theorem 15 in [7].

For the case of exactly two orbits of solutions, we choose a solution S_1 from the first orbit and a solution S_2 from the second orbit. Any other solution S_3 with $S_3 \neq S_1$ and $S_3 \neq S_2$ has a literal $v \in S_3 \setminus S_1$ and a literal $w \in S_3 \setminus S_1$. If $v = w$ we can add a constraint to forbid w and thus forbid S_3 without affecting neither S_2 nor S_1, since w is contained exclusively in solution S_2. If $v \neq w$, we can add a constraint forbidding $\{v, w\}$ without affecting neither S_1 nor S_2, since the pair of literals $\{v, w\} \nsubseteq S_1$ and $\{v, w\} \nsubseteq S_2$. So, a binary realisable SRVR exists that has S_1 and S_2 as orbit representative solutions. □

The first two cases of Theorem 4 subsume the case where there is only one orbit of solutions, like a CSP consisting of an alldifferent constraint. For this case, Puget [12] showed that LLCs reduced to binary constraints suffice. Other CSPs that only have one orbit of solutions include n-queens for $n = 4, 5, 6$ and the graceful graph problem for $K_5 \times P_2$ [1]. Another example that can easily be generalised to a class of problems is the CSP with 4 variables x_1, x_2, x_3, x_4, the domain $\{1, \ldots, 4\}$ and constraints $x_1 \neq x_2$, $x_3 \neq x_4$.

In the reminder of this section, we focus on binary CSPs. First we consider CSPs with a restriction on the number of orbits and then, we consider CSPs with a restriction on the MSC.

6.1 Binary CSPs with Three Orbits of Solutions

In this section, we prove that for binary CSPs with three orbits of solutions, there always is a binary realisable SRVR. We need some more notation. In the microstructure of a binary CSP with a complete set of orbit representative solutions, we call an edge *erasable* if it does not belong to any orbit representative solution in the complete set. Similar to Section 4.2, we set apart certain complete sets of orbit representative solutions. We call a complete set of orbit representative solutions a *max-erasable set*, if the number of erasable edges is maximal. A max-erasable set is a rather special complete set of orbit representative solution for which we show that a binary realisable SRVR exists. We first prove two properties of max-erasable sets.

Proposition 3. *For a* CSP *with three orbits of solutions, let* $R = \{S_1, S_2, S_3\}$ *be a max-erasable set and let* $T \notin R$ *be a solution with the following property. Every pair of literals in* T *is contained in at least one member of* R*. Then,* T *and the orbit representative solution* $S_1 \in R$ *of* $\text{orbit}(T)$ *have at least one pair of literals in common.*

Proof. All pairs of literals of T are contained in some orbit representative solution in R by assumption in the proposition. We prove that a pair of literals in T exists, that is contained in S_1.

Solution S_2 may contain all literals in T apart from, say, literal v. Solution S_3 may contain all of the literals in T apart from, say, literal w. If $v = w$, then any edge $\{\cdot, w\} \subset S_1$, since we have $\{\cdot, w\} \not\subset S_2$ and $\{\cdot, w\} \not\subset S_3$. If $v \neq w$, then the pair of literals $\{v, w\}$ must be contained in either S_1, S_2 or S_3, because of the assumption that all pairs in T are contained in an orbit representative solution. Since $v \notin S_2$ and $w \notin S_3$, we have $\{v, w\} \subseteq S_1$. □

Proposition 4. *With the prerequisites of Proposition 3 the following holds. A literal* $x \in S_1 \setminus T$ *exists that is not contained in* $S_2 \cup S_3$*.*

Proof. For the sake of contradiction we assume that all literals in $S_1 \setminus T$ are contained in either S_2 or S_3. With this assumption, we show the existence of a clique of size $n+1$ in the microstructure. As we noted in Proposition 3, solution S_1 contains at least a pair of literals of T. Assume that $|S_1 \cap T| = 2$, the case $|S_1 \cap T| > 2$ is similar.

We first show, that a literal t in $T \setminus S_1$ exists with $t \in S_2$ and $t \in S_3$. Assume that this is not the case, say, literal $t \in S_2$ and $t \notin S_3$. A literal $s \in T$ exists with $s \notin S_2$ because $T \neq S_2$. The pair of literals $\{s, t\}$, we have $\{s, t\} \subseteq S_2$ because $s \notin S_2$. Furthermore, $\{s, t\} \nsubseteq S_3$ by the assumption $t \in S_2$, $t \notin S_3$ and literal t is chosen in such that $t \notin S_1$. Hence, the pair $\{s, t\}$ is not contained in any orbit representative solution, which is a a contradiction to the assumption that all pair of literals in T are contained in an orbit representative solution.

So, a literal $t \in T \setminus S_1$ exists with $t \in S_2$ and $t \in S_3$. By assumption, all literals in $S_1 \setminus T$ are either contained in S_2 or S_3. Hence, any literal in $S_1 \setminus T$ is adjacent in the microstructure to literal t with $t \in S_2$ and $t \in S_3$. All literals in T, in particular the literals in $T \cap S_1$ are adjacent to literal t because $t \in T$. Hence, $S_1 \cup \{t\}$ is a $n+1$-clique which can never occur in a microstructure of a CSP with n variables. □

We can now prove the main result of this section.

Theorem 5. *A binary realisable* SRVR *exists for any binary* n*-variable* CSP *with three orbits of solutions.*

Proof. By contradiction. Consider a max-erasable set of orbit representative solutions consisting of solutions S_1, S_2 and S_3. We assume that in some orbit of solutions $\text{orbit}(S_1)$, a solution $T \neq S_1$ exists such that all pairs of literals in T are contained in solutions in the max-erasable set, otherwise there is nothing to prove. By Proposition 4, one of the literals, say, $x \in S_1 \setminus T$ is not contained

$S_2 \cup S_3$. Hence, making T the orbit representative solution of orbit(T), we augment the number of erasable pairs of literals which is a contradiction to the assumption that $\{S_1, S_2, S_3\}$ is a max-erasable set. □

The above theorem could probably be generalised to more orbits of solutions, as a more direct use of the symmetries should provide extra leverage for proofs. We leave such generalisations for future work.

6.2 Binary CSPs with Restricted MSCs

So far, we have proved that for CSPs with a restricted number of orbits, there always is a binary realisable SRVR. In general, however, we assume solutions to be unknown to the constraint programmer and hence, our existence results are of limited practical importance. Next, we consider CSPs with a restriction on the MSC without restricting the solutions of the CSP directly.

We define PATH as the class of CSPs whose MSC is a path. We note that any CSP in PATH has domains of size at most 2. The constraints of the CSPs in PATH can be thought of as generalised implications.

Proposition 5. *A binary realisable SRVR exists for any CSP in PATH.*

Proof. We note first that any CSP in PATH that has a variable whose domain is of size 1, admits at most one solution, hence we consider the case where all domains have size 2. We also assume that the CSP has more than 2 literals, otherwise we simply forbid one of the two literals. Let the CSP have variables $x_1, \ldots, x_n$. Without loss of generality, we may assume that each variables has domain $\{1, 2\}$ and that $\{(x_i, 2), (x_{i+1}, 1)\}$ is an edge in the MSC for every $i \in \{1, \ldots, n-1\}$. We note that a path has one non-trivial symmetry. For all $i < \lceil n/2 \rceil$ and $j \in \{1, 2\}$, the symmetry swaps (x_i, j) with $(x_{n-i+1}, j + 1 \bmod 3)$. This symmetry can be thought of as a reflection about the centre of the path. In a CSP in PATH we have the following solutions. The solutions where all variables have value 1 is symmetric to the solution where all variables have value 2. Next, we have a solution, where the first i variables have value 1 and $n - i$ variables have value 2. This solution is symmetric to a solution where the first $n - i$ variables have value 1 and the last i variables have value 2. The CSP has no other solutions.

We forbid the pair of literals $\{(x_1, 2), (x_2, 2)\}$. This removes the solution where all variables have value 2, but leaves the other orbit member. Next, we consider the other solutions where the first i variables have value 1 and the next $n - i$ variables have value 2. If we have $n - i = i$, then we do not add a binary constraint, because in this case the orbit only has one member. If we have $n - i \neq i$, then the orbit has two solutions. We forbid $\{(x_i, 1), (x_{i+1}, 2)\}$. Certainly, this leaves the solution where all variables have value 1. It also does not disallow any other solution from other orbits, but it forbids one member of its own orbit. Hence, a binary realisable SRVR exists for CSPs in PATH. □

Next, we consider CSPs with MSC whose connected components are paths. We denote this class by PATH*.

(a) A connected component corresponding to one variable.

(b) The c-solution is self-symmetric with respect to the p-symmetries.

(c) The c-solution is c-symmetric to the partial solution in Figure 4c.

Fig. 4: Connected components of a MSC that are allowed by the constraints of Proposition 5 as in the proof of Theorem 6. The literals in the c-solutions are depicted as black.

Theorem 6. *A binary realisable SRVR exists for any CSP in PATH***.*

Applying Proposition 5 to every connected component is not enough to prove the above theorem. If the MSC of a CSP consists of connected components as depicted in Figures 4b and 4c, the symmetry that swaps the two connected components is not broken by the constraints presented in the proof of Proposition 5.

Proof. Consider a CSP P whose MSC is a disjoint union of paths. We call a partial solution that is a solution in the connected component a *c-solution*. We distinguish between two types of symmetries. We call symmetries that permute literals within a connected component *p-symmetries*. We call symmetries that permute entire connected components *c-symmetries*. Any symmetry of P is a composition of p- and c-symmetries. As in the proof of Proposition 5, we assume without loss of generality that the domains of the variables are subsets of $\{1, 2\}$ and that the only edges in the MSC are $\{(x_i, 2), (x_{i+1}, 1)\}$. Unlike Proposition 5, we need to consider variables that have a single value in their domains.

We first consider the case of a CSP whose MSC has symmetrically distinct connected component, i.e., P has no c-symmetries. We can rely on the arguments in the proof of Proposition 5. We consider every connected component on its own. For a connected component with an even number of literals greater than 3, we add a constraint as in the proof of Proposition 5, i.e., forbidding $\{(x_i, 2), (x_{i+1}, 2)\}$. The set of constraints we have introduced so far, forbid all solutions that contain no c-solution that is a) self symmetric under the p-symmetries and b) that consists of a variable and its domain (see Figure 4 for examples). By the assumption of connected components not being symmetric, a c-solution that is p-self-symmetric can only be in a solution that has an orbit of size greater than 1, if another c-solution is not p-self-symmetric. Hence, we only need to consider solutions that contain a c-solution on a connected component similar to Figure 4a and consists of, say, literals $(x_i, 1)$ and $(x_i, 2)$. Here, it is easy to show that binary realisable SRVR exists. Any solution that contains $(x_i, 2)$ can be disallowed, since a symmetric equivalent exists that contains $(x_i, 1)$. So, unary constraints suffice.

Next, we consider the case where P has symmetric connected components. In this case, a p-self-symmetric c-solution can be symmetric to a c-solution that a p-symmetry maps to a different c-solution, see Figures 4b and 4c. For this case, we need arguments that are beyond the proof of Proposition 5. The interesting case of p-self-symmetric c-solutions are solutions with an even number of variables and every variable has domain of size 2 as we argue in the following. As soon as a variable has a domain of size 1 in the connected component there can only be one c-solution in the connected component. This case is uninteresting because a p-self-symmetric c-solution is c-symmetric to a p-self-symmetric c-solution. So, we assume to have k symmetric connected components each of which has ℓ variables. Each variable has domain of size 2. We number the connected components and denote variables with two indices, the first for the connected component and the second for the number in which the variable appears in a straight-forward order of the literals in the path. We add the following constraints. Any solution that contains $(x_{i,1+\ell/2}, 2)$ we forbid by simply adding a constraint that does not allow the combination of this literal with $(x_{i-1,1+\ell/2}, 1)$. Let us prove that this constraint does not remove an entire orbit. The literal $(x_{i-1,1+\ell/2}, 1)$ is only contained in the p-self-symmetric c-solution. No other c-solution contains this literal. We note however that we cannot simply remove the literal $(x_{2,1+\ell/2}, 2)$. This would remove an orbit of solutions, namely the orbits where all c-solutions in the symmetric connected components are the same and self-symmetric. Hence, we have shown that a binary realisable SRVR exists. □

For CSPs with a MSC whose connected components are cycles, denoted by CYCLE*, we can prove an analogous result to Theorem 6.

Proposition 6. *A binary realisable SRVR exists for any CSP in CYCLE*.*

Proof. We consider the case of a single connected component first. For an odd number of literals in the MSC, the only case where a solution exists, is the case of a CSP with 1 variable and domain size 3. In this case, we can simply add constraints that forbid 2 literals with unary constraints. A MSC that is an odd cycle of length greater than 3 does not have a solution. For a satisfiable CSP with a MSC that is an even cycle of size $2n$, the CSP has n variables each with domain size 2 and we have one orbit of solutions, so we can rely on Theorem 4.

For CSPs with more than one connected component, we note that we do not have the case of self-symmetric solutions in the connected components as in the proof of Theorem 6. Hence, the arguments for the case of a single connected component suffice to terminate this proof. □

Theorem 6 and Proposition 6 extend naturally to any CSP with a MSC that consists of connected components that are either cycles or paths.

7 Conclusion and Further Work

We considered complete symmetry breaking for the full group of constraint symmetries in this paper. We obtained the following bounds on the arities of symmetry breaking constraints. We showed that no reduction rule for LLCs will succeed in reducing the arity of the LLCs of all CSPs, therefore obtaining an *upper bound* on the arities of LLCs. However, we also showed that constraints of arity $\lfloor n/2 \rfloor + 1$ always exist that provide complete symmetry breaking. The latter is an existence result and does not come with an algorithm. Further work could try to find variable ordering strategies, such that reduction rules will always reduce the arity of LLCs. We found a *lower bound* on the arities of symmetry breaking constraints: we proved that binary CSPs exist where complete symmetry breaking constraints must be at least ternary. However, we also presented various special cases of CSPs where binary constraints break all symmetries.

Acknowledgements

The author thanks Barbara Smith for discussions and Marc van Dongen for the same as well as insightful comments during the preparation of this paper.

References

1. D. Cohen, P. Jeavons, C. Jefferson, K. Petrie, and B. Smith. Symmetry definitions for constraint satisfaction problems. *Constraints 11*, 2006.
2. J. Crawford, M. Ginsberg, E. Luks, and A. Roy. Symmetry-breaking predicates for search problems. In *Principles of Knowledge Representation and Reasoning (KR '96)*. Morgan Kaufmann, 1996.
3. A. Frisch, B. Hnich, Z. Kiziltan, I. Miguel, and T. Walsh. Global constraints for lexicographic orderings. In *Principles and Practice of Constraint Programming (CP 02)*, 2002.
4. A. M. Frisch and W. Harvey. Constraints for breaking all row and column symmetries in a three-by-two matrix. In *In Proceedings of SymCon'03*, 2003.
5. I. P. Gent, K. E. Petrie, and J.-F. Puget. Symmetry in constraint programming. In F. Rossi, P. van Beek, and T. Walsh, editors, *Handbook of Constraint Programming*, pages 329–376. Elsevier, 2006.
6. A. Grayland, C. Jefferson, I. Miguel, and C. Roney-Dougal. Minimal ordering constraints for some families of variable symmetries. *Annals of Mathematics and AI*, 2009.
7. T. Januschowski, B. M. Smith, and M. R. C. van Dongen. Foundations of symmetry breaking revisited. In *Proceedings of SymCon'09*, 2009.
8. V. Kaibel and M. E. Pfetsch. Packing and partitioning orbitopes. *Mathematical Programming*, 114, Number 1 / July, 2008:1–36, 2008.
9. Y. C. Law and J. H. Lee. Symmetry breaking constraints for value symmetries in constraint satisfaction. *Constraints*, 11(2-3):221–267, 2006.
10. H. Öhrmann. Breaking symmetries in matrix models. Master's thesis, Dept. Information Technology, Uppsala University, 2005.
11. J.-F. Puget. On the satisfiability of symmetrical constrained satisfaction problems. In *Methodologies for Intelligent Systems*, pages 350–361, London, UK, 1993. Springer-Verlag.
12. J.-F. Puget. Breaking symmetries in all-different problems. In *International Joint Conferences on Artificial Intelligence*, pages 272–277, 2005.
13. T. Walsh. General symmetry breaking constraints. In *Principles and Practice of Constraint Programming – CP 2006*, volume 4204/2006 of *Lecture Notes in Computer Science 4204, 2006*, pages 650–664, 2006.