

Small Sets of Ordering Constraints for some Families of Variable Symmetries

Andrew Grayland¹, Chris Jefferson², Ian Miguel¹, Colva Roney-Dougal³

¹School of Comp. Sci., Univ. of St Andrews, UK. {andyg,ianm}@cs.st-and.ac.uk

²Oxford Univ. Computing Laboratory, UK. Chris.Jefferson@comlab.ox.ac.uk

³School of Maths and Statistics, Univ. of St Andrews, UK. colva@mcs.st-and.ac.uk

Abstract. Existing methods of breaking all variable symmetries by adding lexicographic ordering constraints can require a factorial number of constraints. This adds an unacceptable overhead to the solving process. In certain cases, for example when the variables are constrained to take distinct values, a reduction to a linear set of constraints is possible. This paper studies some commonly-occurring families of groups and shows how the set of ordering constraints can be reduced in each case.

1 Introduction

Constraint Programming supports the solution of a combinatorial problem in two stages. First, by characterising or *modelling* it as a *constraint satisfaction problem* (CSP): a finite set of decision variables, each with a finite set of potential values, and a set of constraints on the allowed assignments of values to the variables. Second, a constraint solver is used to search for *solutions*: assignments to the decision variables that satisfy all the constraints. Constraint models often contain *symmetries*: bijections from (non-)solutions to (non-)solutions. These symmetries can be exploited by restricting the search for a solution to one member of each equivalence class (*symmetry breaking*), dramatically reducing search.

One symmetry breaking method is to add constraints to the model. Crawford *et al* [2] describe one such approach called *lex-leader*: one member of each equivalence class is designated as lexicographically least, and a set of lexicographic ordering constraints are added to preclude all other members of that class. The disadvantage of this method is that, for a CSP with n variables, it can produce $n!$ lexicographic ordering constraints. The overhead of adding this number of constraints to the CSP usually outweighs the benefit of breaking the symmetry.

In many cases, this large set of constraints can be reduced to a much smaller set that still breaks all the symmetry [3, 5]. However, these general reduction methods are themselves prohibitively costly. Puget [7] has identified a special case, where each variable must be assigned a distinct value, in which the set of ordering constraints collapses to just $n - 1$ binary inequalities. This paper follows somewhat in this vein. It considers the mathematical *group* that describes certain symmetries, the associated set of lexicographic ordering constraints necessary to break those symmetries, and how that set can be reduced.

2 Background

A finite-domain constraint satisfaction problem comprises: a finite set of variables \mathcal{X} ; for each variable $x \in \mathcal{X}$, a finite set of values (its domain); and a finite set

\mathcal{C} of constraints on the variables. Each constraint $c \in \mathcal{C}$ is defined over a subset $\mathcal{X}' \subseteq \mathcal{X}$ by a subset of the Cartesian product of the domains of the members of \mathcal{X}' , giving the set of allowed combinations of values. A *complete assignment* maps every variable in a given CSP to a member of its domain.

A *variable symmetry* of a CSP is a bijection $f : \chi \rightarrow \chi$ of the set of variables such that $\{\langle v_i, a_i \rangle : 1 \leq i \leq n\}$ is a solution if and only if $\{\langle f(v_i), a_i \rangle : 1 \leq i \leq n\}$ is a solution.

Any group can be represented by a set G of bijections from a set X to itself (or *permutations* of the set X), such that G is closed under composition of functions and inversion. The groups we are interested in are sets of variable symmetries of the CSP. The symmetric group, S_n , is the group whose elements are the set of all possible bijective functions mapping $\chi \rightarrow \chi$, $|\chi| = n$.

Having identified a symmetry within a model we then work out which constraints to add in order to break it, using the lex-leader method. We first define an ordering on the decision variables.¹ We then add constraints which order the values these variables can assume. Take the elements of the symmetric group S_3 as an example. Here the permutation (ACB) means A maps to itself, and B and C map to each other.

$$(ABC), (BAC), (ACB), (CBA), (BCA), (CAB)$$

We take our decision variable ordering to be ABC and post constraints such that only one permutation of the values assigned to these variables in each equivalence class can satisfy these constraints. Lex-Leader gives us the following constraints, which are proven to break S_3 :

$$ABC \leq_{lex} BAC, ABC \leq_{lex} ACB, ABC \leq_{lex} CBA \\ ABC \leq_{lex} BCA, ABC \leq_{lex} CAB$$

There is one constraint per nontrivial permutation of S_3 . In general, there are $n!$ permutations for the group S_n and so $(n! - 1)$ n -ary constraints are produced by the lex-leader method to break all symmetries.

3 Reducing the Number of Ordering Constraints

Frisch and Harvey [3] describe two rules to reduce the number and arity of constraints whilst maintaining complete symmetry breaking:

1. If we have a constraint C of the form $\alpha X \beta \leq_{lex} \gamma Y \delta$, and $\alpha = \gamma$ logically implies $X = Y$ then we may replace it with $\alpha \beta \leq_{lex} \gamma \delta$.
2. If we have a set of constraints C of the form $C' \cup \{\alpha \beta \leq_{lex} \gamma \delta\}$, and $C' \cup \{\alpha = \gamma\}$ logically implies $\beta \leq_{lex} \delta$, then we may replace C with $C' \cup \{\alpha \leq_{lex} \gamma\}$.

Rule 1 reduces our set of lex constraints to: $A \leq_{lex} B, B \leq_{lex} C, A \leq_{lex} C, AB \leq_{lex} BC, AB \leq_{lex} CA$. Application of rule 2 simplifies the constraints

¹ Note that recent research suggests that the exact element chosen as the lex-least can affect the search tree quite considerably [8].

further to: $A \leq_{lex} B, B \leq_{lex} C$. Simplifying a factorial number of constraints using these rules is expensive. In practice this number of operations has proved infeasible as much of the time saved by breaking the symmetries is re-introduced in this pre-processing stage [5]. We can derive a lower bound on the number of binary inequality constraints produced by this process:

Theorem 1. *Given a CSP with n decision variables with a transitive² group of variable symmetries the minimum number of binary \leq constraints required to remove all but one member of each equivalence class is $n - 1$.*

Proof. Any constraint graph with $n-2$ binary constraints for a problem involving n decision variables is disconnected. Assume without loss of generality that X_1 is the minimal variable with respect to the ordering. Let X_i be a decision variable that is not connected to X_1 , and such that X_i is minimal in its component of the constraint graph. Consider the full assignment which assigns $X_i = a$ where a is minimal in the domain of X_i , and $X_j = b$ for $j \neq i$, where $b > a$. Since there is a symmetry mapping X_i to X_1 there is a symmetry mapping this full assignment to a full assignment with $X_1 = a$, which is lexicographically less. Thus the equivalence class under the group has size greater than 1, and $n - 2$ binary constraints will not suffice.

4 Small Sets of Lex Constraints for some Families of Groups

We now consider some specific families of groups. If all elements of a group G can be written as powers of some fixed $g \in G$ then G is *cyclic*. Every permutation of a group can be written as permutations interchanging pairs of points (also called transpositions). A permutation is even if it can be written as a product of an even number of transpositions. The subgroup of S_n that contains all of the even permutations is A_n , the *alternating group*.

Theorem 2. *Let C be a CSP with $\chi = \{X_1, \dots, X_n\}$. If the symmetry group of C is one of C_n , A_n or S_n on variables then a complete set of symmetry breaking constraints is:*

C_n	$X_1 \leq_{lex} X_2, X_1 X_2 \leq_{lex} X_3 X_4, X_1 X_2 X_3 \leq_{lex} X_4 X_5 X_6, \dots$ $X_1 X_2 \dots X_{n/2+1} \leq_{lex} X_{n/2+1} \dots X_n X_1 X_2, \dots$ $X_1 X_2 \dots X_{(n-1)/2} X_{(n+1)/2} \leq_{lex} X_{(n+3)/2} \dots X_n X_1, \dots$ $X_1 X_2 \dots X_{n-1} \leq_{lex} X_n X_1 \dots X_{n-2}$	n even n odd
A_n	$X_i \leq_{lex} X_{i+1}, X_{n-2} \leq_{lex} X_n$ $X_i X_{n-1} \leq_{lex} X_{i+1} X_n, X_{n-2} X_{n-1} \leq X_n X_{n-2}$	$1 \leq i \leq n - 2$ $1 \leq i \leq n - 3$
S_n	$X_i \leq_{lex} X_{i+1}$	$1 \leq i \leq n - 1$

Figure 1a defines the Circular (or Modular) Golomb Ruler problem. Two solutions to the instance of this problem where n is 7 and m is 3 are shown in Figure 1b. Clearly, these solutions are symmetric: one can be obtained from the other via rotation. This problem has cyclic symmetry.

² any variable can map to any other.

- a) Circular Golomb Ruler Problem: b)
 Given a circle with circumference n , place m ticks at integer points around the circle such that all inter-tick distances along the circumference are distinct. (n and m are both positive integers).

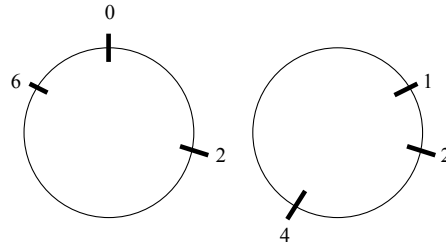


Fig. 1. Specification of the Circular Golomb Ruler problem. Symmetric solutions to the length 7, 3-tick Circular Golomb Ruler problem.

The symmetry group in the above problem, $n = 7$ $m = 3$, is C_3 . The symmetry breaking constraints required to order the set of ticks, $T = \{t_1, t_2, t_3\}$ are $t_1 \leq_{lex} t_2$, and $t_1 t_2 \leq_{lex} t_3 t_1$.

Often, symmetry groups can be built up out of smaller, easier-to-describe groups. When this occurs we take the product of several groups.

Definition 1. Let $G \leq \text{Sym}(\Omega)$ and $H \leq \text{Sym}(\Delta)$ be groups, with Ω and Δ disjoint sets. The direct product of G and H , written $G \times H$, is the set $\{(g, h) : \forall g \in G, h \in H\}$, with coordinatewise multiplication. Elements of $G \times H$ permute the set $\Omega \cup \Delta$ as follows: $(g, h)(x) = g(x)$ if $x \in \Omega$, and $(g, h)(x) = h(x)$ if $x \in \Delta$.

Consider now the problem of finding two distinct circular Golomb rulers. The symmetry in this problem is the direct product of the two cyclic groups.

Theorem 3. Given groups G and H the set of complete symmetry breaking constraints of $G \times H$ is $L_G \cup L_H$, where L_G is the set of lex constraints for G and L_H is the set of lex constraints for H .

For example take two sets of ticks, $T = \{t_1, t_2, t_3\}$ and $T' = \{t'_1, t'_2, t'_3\}$, defining two distinct circular golomb rulers for $n = 7$ $m = 3$. The lex constraints required to break the symmetry are $t_1 \leq_{lex} t_2$, $t'_1 \leq_{lex} t'_2$, $t_1 t_2 \leq_{lex} t_3 t_1$ and $t'_1 t'_2 \leq_{lex} t'_3 t'_1$.

Another commonly arising way of combining two groups is the *imprimitive wreath product*.

Definition 2. Let $G \leq \text{Sym}(n)$ and $H \leq \text{Sym}(k)$. The imprimitive wreath product of G and H , denoted $GWrH$, is a subgroup of $\text{Sym}(nk)$. It acts on k copies of the set of size n on which G acts. We have $GWrH = \{h(g_1, \dots, g_k) : h \in H, g_i \in G\}$, and these elements permute the set $\{(i, j) : 1 \leq i \leq n, 1 \leq j \leq k\}$ by $h(g_1, \dots, g_k)(i, j) = (g_j(i), h(j))$.

An example of where this occurs is the social golfers problem. This requires partitioning a set of golfers into equal groups in each week of a tournament such that no golfer plays any other more than once. The symmetric group interchanges each of the k golfers in each group, and the symmetric group interchanges each of the l groups in a week, so $S_k Wr S_l$ acts on the golfers in each week.

Suppose we have a set A of symmetry breaking constraints for a group G acting on variables X_1, \dots, X_n , and a set B of symmetry breaking constraints for a group H acting on variables Y_1, \dots, Y_k . Then $GWrH$ acts on a set of nk variables, X_{ij} , with $1 \leq i \leq n$ and $1 \leq j \leq k$. The group $GWrH$ has size $|G|^k \times |H|$, so writing down one constraint for each nontrivial group element is impractical. We now show how to reduce this to $k|A| + |B|$ constraints.

We first post $k|A|$ constraints, namely a copy of A on X_{ij} for each value of j . That is, we lex order each block of variables with respect to G . The arity of these constraints is unchanged.

We then restate the constraints from B so that instead of being statements about the values of sequences of Y s, they are statements about the values of sequences of $X_{1j}X_{2j}\dots X_{nj}$ s. For example, if we previously had a constraint $Y_1 \leq Y_2$ we would replace that with the constraint $X_{11}X_{21}\dots X_{n1} \leq X_{21}X_{22}\dots X_{n2}$. This results in $|B|$ constraints, each of arity n times their original arity.

Theorem 4. *This set of constraints is complete.*

The full symmetry group of the golomb rulers problem is C_3WrS_2 . This is because each circular ruler can swap with any other. To break these symmetries we use the constraints.

$$t_1 \leq_{lex} t_2 \quad t'_1 \leq_{lex} t'_2 \quad t_1 t_2 \leq_{lex} t_3 t_1 \quad t'_1 t'_2 \leq_{lex} t'_3 t'_1 \quad t_1 t_2 t_3 \leq_{lex} t'_1 t'_2 t'_3$$

5 Conclusion

This paper has discussed symmetry breaking in CSPs by adding lexicographic ordering constraints. Given the huge number of such constraints needed in general to break all symmetry, and the intractability of the general methods of reducing this number, we focussed on a number of special cases and showed how the number of ordering constraints necessary in each case can be reduced. In future, we will integrate this work into the automated modelling system CONJURE [4] so that it is able to break symmetry efficiently as it is introduced.

Acknowledgements Andrew Grayland is supported by an EPSRC CASE studentship, sponsored by Microsoft Research. Ian Miguel is supported by a Royal Academy of Engineering/EPSRC Research Fellowship. Colva Roney-Dougal is partly funded by the Nuffield Foundation

References

1. D. Cohen, P. Jeavons, C. Jefferson, K. E. Petrie and B.M. Smith. Symmetry Definitions for Constraint Satisfaction Problems *Proc. CP*, 17-35, 2005
2. J. Crawford, M. Ginsberg, E. Luks and A. Roy. Symmetry-Breaking Predicates for Search Problems *Proc. KR*, 148-159, 1996.
3. A. M. Frisch and W. Harvey. Constraints for Breaking All Row and Column Symmetries in a Three-by-Two-Matrix. *Proc. Symcon*, 2003.
4. A. M. Frisch, C. Jefferson, B. Martinez Hernandez and I. Miguel. The Rules of Constraint Modelling. *Proc. IJCAI*, 109-116, 2005.
5. H. Öhrman. *Breaking Symmetries In Matrix Models*. Masters Thesis, Dept. Information Technology, Uppsala University, 2005.
6. Jean-François Puget. On The Satisfiability of Symmetrical Constraint Satisfaction Problems. *Proc. ISMIS*, 350-361, 1993.
7. Jean-François Puget. Breaking Symmetries In All Different Problems. *Proc. IJCAI*, 272-277, 2005.
8. B. M. Smith. Sets of Symmetry Breaking Constraints. *Proc. Symcon*, 2005.