Minimal Ordering Constraints for Some Families of Variable Symmetries

Andrew Grayland
Ian Miguel
Colva Roney-Dougal

Overview

- Symmetry
- Lex-Leader
- Reducing Ordering Constraints
- General Formulae for some families of variable symmetries
  - Symmetric
  - Cyclic
  - Alternating
  - Dihedral
- Building blocks for more complex symmetries

Variable Symmetry

- A permutation of variables such that the set of (non-)solutions remains unchanged.
- Can be expressed as a mathematical group.
- Symmetric group and Cyclic group used throughout.
- Symmetric Group: Any variable can exchange with any other any number of times and maintain the set of (non-)solutions.
- Cyclic group described later.

Lex-Leader Constraints

- A static symmetry breaking technique.
- Break variable symmetries in CSPs.
- Easy to use.
- Very popular.
- Widely implemented in solvers.

Using Lex-Leader

- Decide on a canonical variable ordering.
- Decide on a value ordering.
- Identify the variable symmetries in a problem.
- Add one constraint per variable symmetry which orders the possible values assigned to each variable.

Using Lex-Leader

- Problem – Find three positive integers such that their sum is 10.
- Three variables: $\{x_1,x_2,x_3\}$
- Each with domain: {1,2,3,4,5,6,7,8}
- One constraint: $x_1+x_2+x_3 = 10$

Using Lex-Leader

- Decide on a canonical variable ordering.
- Decide on a value ordering.
- Identify the variable symmetries in a problem.
- Add one constraint per variable symmetry which orders the possible values assigned to each variable.

Using Lex-Leader

- Three variables – $\{x_1,x_2,x_3\}$
- Choose the ordering $x_1x_2x_3$.
- The ordering is not important here.

Using Lex-Leader

- Decide on a canonical variable ordering.
- Decide on a value ordering.
- Identify the variable symmetries in a problem.
- Add one constraint per variable symmetry which orders the possible values assigned to each variable.

Using Lex-Leader

- Positive Integer Ordering
- Least to greatest.

Using Lex-Leader

- Decide on a canonical variable ordering.
- Decide on a value ordering.
- Identify the variable symmetries in a problem.
- Add one constraint per variable symmetry which orders the possible values assigned to each variable.

Using Lex-Leader

- Symmetry – All variables can be swapped with any other any number of times.
- Symmetric Group.
  - $x_1x_2x_3 \rightarrow x_1x_2x_3$ ()
  - $x_1x_2x_3 \rightarrow x_1x_3x_2$ (23)
  - $x_1x_2x_3 \rightarrow x_2x_1x_3$ (12)
  - $x_1x_2x_3 \rightarrow x_2x_3x_1$ (123)
  - $x_1x_2x_3 \rightarrow x_3x_1x_2$ (132)
  - $x_1x_2x_3 \rightarrow x_3x_2x_1$ (13)

Using Lex-Leader

- Decide on a canonical variable ordering.
- Decide on a value ordering.
- Identify the variable symmetries in a problem.
- Add one constraint per variable symmetry which orders the possible values assigned to each variable.

Using Lex-Leader

- $x_1x_2x_3 \leq_{lex} x_1x_2x_3$ $x_1x_2x_3$ → $x_1x_2x_3$ ()
- $x_1x_2x_3 \leq_{lex} x_1x_3x_2$ $x_1x_2x_3$ → $x_1x_3x_2$ (23)
- $x_1x_2x_3 \leq_{lex} x_2x_1x_3$ $x_1x_2x_3$ → $x_2x_1x_3$ (12)
- $x_1x_2x_3 \leq_{lex} x_2x_3x_1$ $x_1x_2x_3$ → $x_2x_3x_1$ (123)
- $x_1x_2x_3 \leq_{lex} x_3x_1x_2$ $x_1x_2x_3$ → $x_3x_1x_2$ (132)
- $x_1x_2x_3 \leq_{lex} x_3x_2x_1$ $x_1x_2x_3$ → $x_3x_2x_1$ (13)

How Lex-Constraints work

- $x_1x_2x_3 \leq_{lex} x_2x_3x_1$
- Read:
  - $x_1 \leq x_2$
  - If $x_1 = x_2$ then
    - $x_2 \leq x_3$
    - If $x_2 = x_3$ then
      - $x_3 \leq x_1$

Problem With Lex-Leader

- There is always one new constraint per permutation of the symmetry group.
- With the symmetric group of order n this is n! new constraints.
- These can often be prohibitively costly.
- Can we make it more efficient?

Lex-Leader and alldifferent

- Puget defines lex-leader in the presence of alldifferent.
- We need only ever consider the first non-trivial condition of each constraint.
- At most n-1 binary constraints.

Possible Solutions

- $x_1x_2x_3 \leq_{lex} x_1x_2x_3$
- $x_1x_2x_3 \leq_{lex} x_1x_3x_2$ $x_2 < x_3$ $\mathbf{x_2 < x_3}$
- $x_1x_2x_3 \leq_{lex} x_2x_1x_3$ $x_1 < x_2$ $\mathbf{x_1 < x_2}$
- $x_1x_2x_3 \leq_{lex} x_2x_3x_1$ $x_1 < x_2$
- $x_1x_2x_3 \leq_{lex} x_3x_1x_2$ $x_1 < x_3$
- $x_1x_2x_3 \leq_{lex} x_3x_2x_1$ $x_1 < x_3$

Reduction Rules

- What if we do not have alldifferent?
- Reduction Rules
- Remove conditions that have previously been implied or stated.

Rule 1

- Inference over a single constraint.
- If a condition is implied to be equal elsewhere in the same constraint, that condition can be removed.

Rule 1

- $\mathbf{x_1x_2x_3} \leq_{lex} \mathbf{x_1x_2x_3}$
- $\mathbf{x_1}x_2\mathbf{x_3} \leq_{lex} \mathbf{x_1}x_3\mathbf{x_2}$ $\quad x_2 \leq x_3$
- $x_1\mathbf{x_2x_3} \leq_{lex} x_2\mathbf{x_1x_3}$ $\quad x_1 \leq x_2$
- $x_1x_2\mathbf{x_3} \leq_{lex} x_2x_3\mathbf{x_1}$ $\quad x_1x_2 \leq x_2x_3$
- $x_1x_2\mathbf{x_3} \leq_{lex} x_3x_1\mathbf{x_2}$ $\quad x_1x_2 \leq x_3x_1$
- $x_1\mathbf{x_2x_3} \leq_{lex} x_3\mathbf{x_2x_1}$ $\quad x_1 \leq x_3$

Rule 2

- Inference over the whole set of lex-constraints.
- Assume equality in the more significant pairs in any one constraint.
- If the least significant pair of that constraint is implied by the remaining constraints it can be removed.

Rule 2

- $x_2 \leq x_3$ $x_2 \leq x_3$
- $x_1 \leq x_2$ $x_1 \leq x_2$
- $x_1x_2 \leq_{lex} x_2x_3$ $x_1x_2 \leq_{lex} x_2x_3$
- $x_1x_2 \leq_{lex} x_3x_1$ $x_1x_2 \leq_{lex} x_3x_1$
- $\mathbf{x_1 \leq x_3}$

$x_1 \leq x_2$ and $x_2 \leq x_3$ therefore $x_1 \leq x_3$.

Rule 2

- $x_2 \leq x_3$
- $x_1 \leq x_2$
- $x_1 x_2 \leq_{lex} x_2 x_3$
- $x_1 \mathbf{x_2} \leq_{lex} x_3 \mathbf{x_1}$

$x_2 \leq x_3$

$x_1 \leq x_2$

$x_1 x_2 \leq_{lex} x_2 x_3$

$\mathbf{x_1 \leq x_3}$

Assume $x_1 = x_3$.

$x_2 \leq x_3$ therefore $x_2 \leq x_1$.

Remove $x_1 \leq x_3$ as before.

Rule 2

- $x_2 \leq x_3$ $\quad$ $x_2 \leq x_3$
- $x_1 \leq x_2$ $\quad$ $x_1 \leq x_2$
- $x_1\mathbf{x_2} \leq_{lex} x_2\mathbf{x_3}$ $\quad$ $\mathbf{x_1 \leq x_2}$

Assume $x_1 = x_2$.

$x_2 \leq x_3$.

Remove duplicate $x_1 \leq x_2$.

Rule 1,2 Minimal Ordering Constraints

- $\mathbf{x_1 \leq x_2}$ $\mathbf{x_2 \leq x_3}$
- Minimal and Complete.
- n-1 constraints for the symmetric group of order n.

Rule 3

- Inference over the whole set of lex-constraints.
- Assume equality in the more significant pairs in any one constraint.
- If the pair under consideration in that constraint is implied by the remaining constraints to be equal then it can be removed.
- We can construct artificial examples where rule 3 is stronger than rule 2 and rule 1.
- Open Question: Is there a non-contrived symmetry where the lex-constraints can be reduced further by rule 3?

Problem With Reduction Algorithm

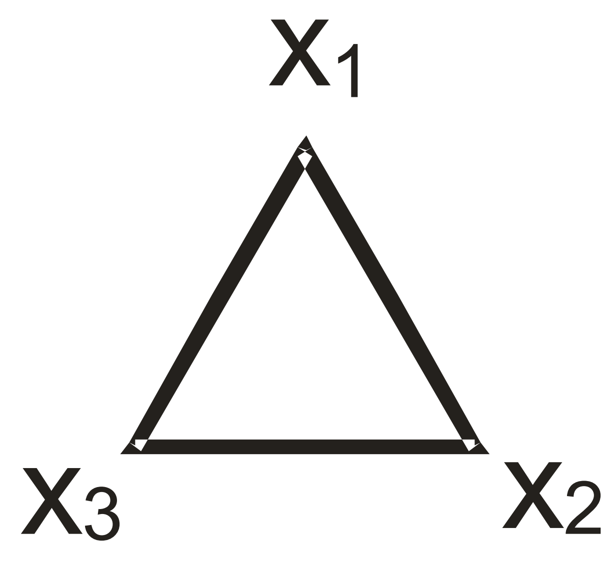
- Reduction algorithm can be expensive; Arbitrary inference on a full set of ordering constraints.
- Minimal Constraints in linear time?

General Formulae

- General Formulae for the production of reduced lex-constraints for four families of symmetries:
  - Symmetric Groups
  - Alternating Groups
  - Cyclic Groups
  - Dihedral Groups
- Minimal: No further reductions possible under reduction rules.
- Complete: Equivalent to lex-constraints.

Cyclic Groups

- Half the symmetries of a regular n sided polygon.
- Only the rotational symmetry.
- The cyclic group of order 3 (An equilateral triangle):
  - $x_1x_2x_3 \rightarrow x_1x_2x_3$ ()
  - $x_1x_2x_3 \rightarrow x_2x_3x_1$ (123)
  - $x_1x_2x_3 \rightarrow x_3x_1x_2$ (132)

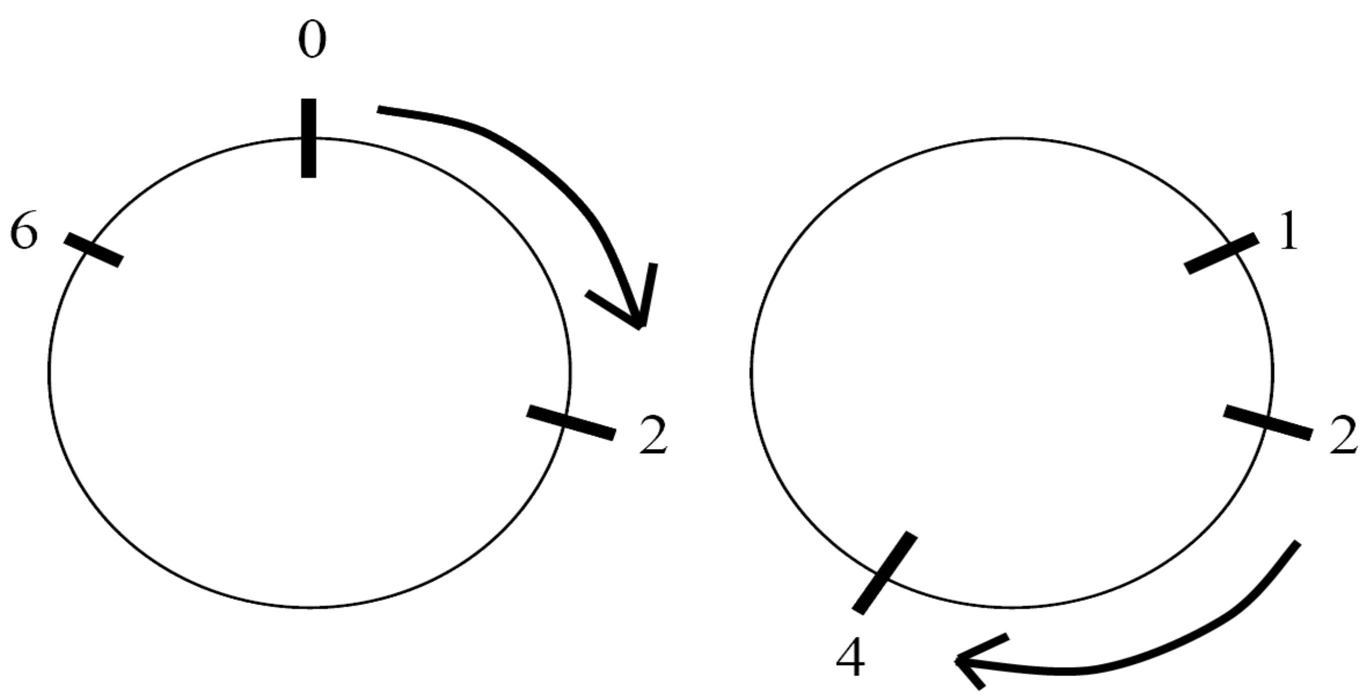


Cyclic Group of Order 3

- Lex-constraints:
  - $x_1x_2x_3 \leq_{lex} x_1x_2x_3$ $\quad x_1x_2x_3 \rightarrow x_1x_2x_3$ $\quad ()$
  - $x_1x_2x_3 \leq_{lex} x_2x_3x_1$ $\quad x_1x_2x_3 \rightarrow x_2x_3x_1$ $\quad (123)$
  - $x_1x_2x_3 \leq_{lex} x_3x_1x_2$ $\quad x_1x_2x_3 \rightarrow x_3x_1x_2$ $\quad (132)$
- Minimal ordering constraints.
  - $x_1 \leq_{lex} x_2$
  - $x_1x_2 \leq_{lex} x_3x_1$
- General Formula in paper.

Circular Golomb Ruler Problem

- 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 > 0 and m > 0.
- Example: n = 7 m = 3



Circular Golomb Ruler Problem

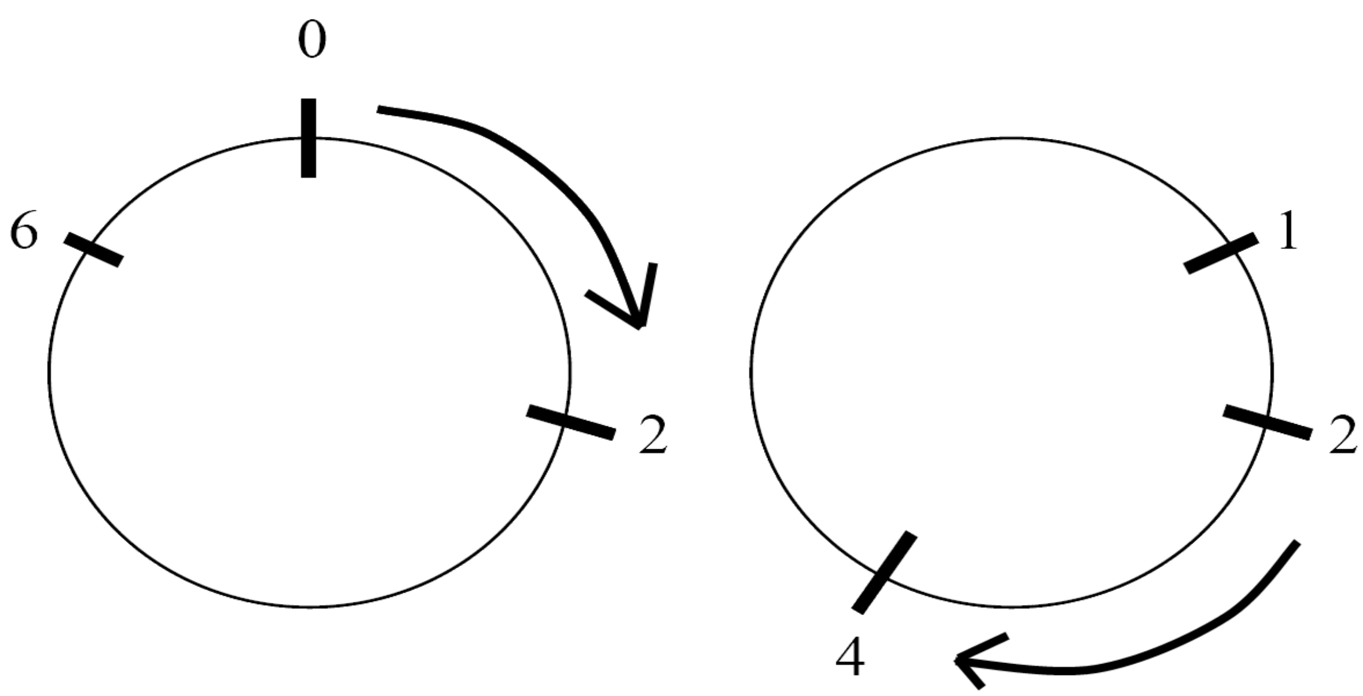
- Clearly this problem has cyclic group symmetry.
- Modeled as n binary variables $\{x_1,...,x_n\}$
  - Sum($\{x_1,...,x_n\}$) = m
  - $x_i - x_j = z_{ij}$, $1 \le i \le n$, $1 \le j \le n$, i != j, z in Z
  - Alldifferent(Z)
- Note: This problem has dihedral symmetry, but for the purposes of demonstration we will only consider the cyclic symmetry.

Recap

- We now have formulae to produce minimal ordering constraints for some groups.
- In practice these perform well against lex-leader.
- The real bonus is now using these minimal formulae as building blocks for much more complex symmetries.

Combining Groups

- Consider now the problem of finding two unique rulers, $\{x_1,...,x_n\}$ and $\{y_1,...,y_n\}$.
- The two rulers are freely interchangeable
- Symmetric group symmetry.



Combining Groups

- Complex symmetries can be made by combining much simpler groups.
- Imprimitive Wreath Product of a symmetric group and a cyclic group.
- 7 units around each ruler, 2 rulers.
  - $n = 7, r = 2$
- $C_n \, \mathit{Wr} \, S_r$

Combining Groups

- Use the cyclic group reduced-lex formula to order the ticks on each ruler.
  - $x_1 \leq_{lex} x_2$
  - $x_1x_2 \leq_{lex} x_3x_4$
  - $x_1x_2x_3 \leq_{lex} x_4x_5x_6$
  - $x_1x_2x_3x_4 \leq_{lex} x_5x_6x_7x_1$
  - $x_1x_2x_3x_4x_5 \leq_{lex} x_6x_7x_1x_2x_3$
  - $x_1x_2x_3x_4x_5x_6 \leq_{lex} x_7x_1x_2x_3x_4x_5$
- Similarly for $\{y_1,\ldots,y_n\}$

Combining Groups

- Next use the symmetric group reduced lex-formula to order the two rulers, substituting the orderings chosen for each for the decision variables in the general formula.
  - $x_1x_2x_3x_4x_5x_6x_7 \leq_{lex} y_1y_2y_3y_4y_5y_6y_7$

Building Blocks

- Given any two minimal sets of ordering constraints for two groups we can construct a minimal set of ordering constraints for the direct product and the imprimitive wreath product of those two groups.
- Easily extensible to new groups.

Conclusion

- General formulae for construction of minimal ordering constraints in linear time.
- Minimal formulae for new groups can be slotted into these combination formulae with no modifications.
- Any solver which implements lex-constraints can work with the reduced constraints.
- Testing shows method works well in practice.

Future Work

- Extend general formulae to break matrix symmetries.
- Automated decomposition of groups returned from symmetry tools such as NAUTY/SAUCY.
- Enable automated symmetry breaking in automated constraint modeling, e.g. CONJURE.