Symmetries in CSP

Elena Sherman

UNL, CSE

April 18, 2009

Table of contents

Contents

What is Symmetry?

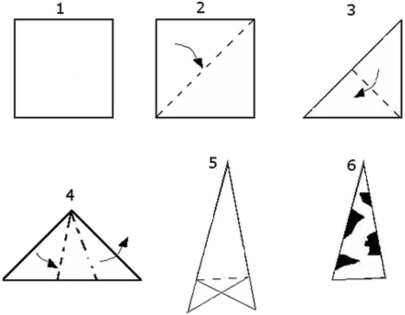
Symmetry

- Defined as "patterned self-similarity".
- Generated by a transformation $\mathcal{S}$ of an object O_1 into O_2.
- $\mathcal{S}(O_1)$ is not distinguishable from O_2.
- Common $\mathcal{S}$ are translation, rotation and reflection.

Crafting a Paper Snowflake

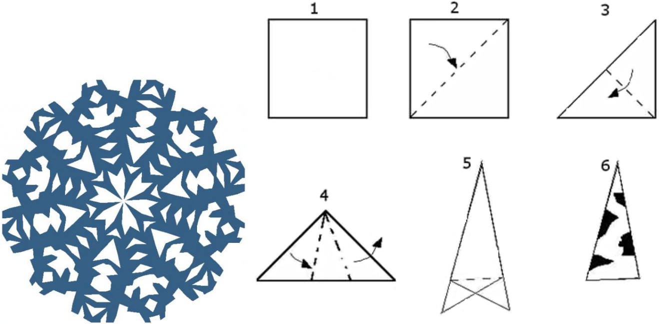
How to cut out a snowflake from a piece of paper?

Crafting a Paper Snowflake



How to cut out a snowflake from a piece of paper?

Crafting a Paper Snowflake



How to cut out a snowflake from a piece of paper?
In general biological science problems have many geometric symmetries.

Why is Symmetry?

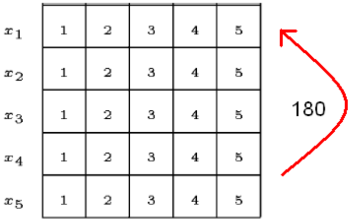
- CSP=$(V, D, C) \in NPC$, but $\exists$ islands of tractability.
- Using the structure of CSP to reduce complexity, or to reduce the problem size.
- Symmetry can occur in V, D and C ex. ALL-DIFF constraint.
- CSP's elements that are symmetric under $\mathcal{S}$ create an equivalence class.
- Property detected in one element of an equivalent class can be generalized to all elements of that class. Ex. $D = \{1, 2, 3, 4, 5, 6, 7\} \Rightarrow D = \{[2, 4, 6], [3, 5, 7]\}$.

Contents

5-queens Symmetry Example $\mathcal{S}$ = 180 Rotation

x_1	1	2	3	4	5
x_2	1	2	3	4	5
x_3	1	2	3	4	5
x_4	1	2	3	4	5
x_5	1	2	3	4	5

5-queens Symmetry Example $\mathcal{S}$ = 180 Rotation



- Rotate by 180 degrees.

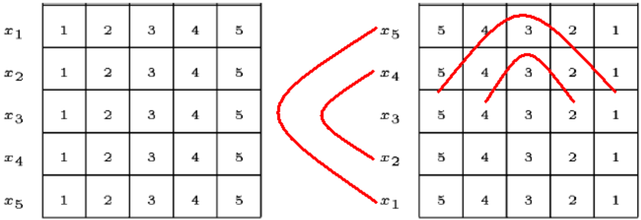
5-queens Symmetry Example $\mathcal{S}$ = 180 Rotation

x_1	1	2	3	4	5
x_2	1	2	3	4	5
x_3	1	2	3	4	5
x_4	1	2	3	4	5
x_5	1	2	3	4	5

x_5	5	4	3	2	1
x_4	5	4	3	2	1
x_3	5	4	3	2	1
x_2	5	4	3	2	1
x_1	5	4	3	2	1

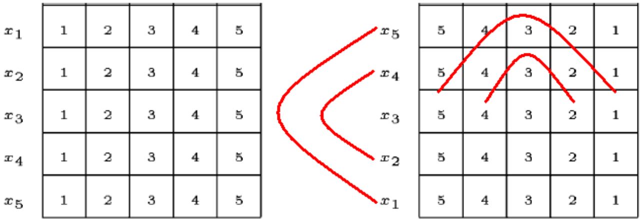
- Rotate by 180 degrees.

5-queens Symmetry Example $\mathcal{S} = 180$ Rotation



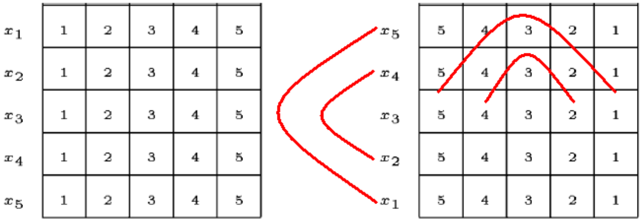
- Rotate by 180 degrees.
- x_1 exchanges with x_5 and x_2 with x_4.
- New domains $\theta(val) = 6 - val$ for each x_i.
- Equivalence classes:
  - Variables $\{x_1, x_2\}$, $\{x_2, x_4\}$ and $\{x_3\}$.
  - Values $\{1, 5\}$, $\{2, 4\}$, $\{3\}$.

5-queens Symmetry Example $\mathcal{S}$ = 180 Rotation



- Rotate by 180 degrees.
- x_1 exchanges with x_5 and x_2 with x_4.
- New domains $\theta(val) = 6 - val$ for each x_i.
- Equivalence classes:
  - Variables $\{x_1, x_2\}$, $\{x_2, x_4\}$ and $\{x_3\}$.
  - Values $\{1, 5\}$, $\{2, 4\}$, $\{3\}$.
- Reflection about the horizontal axis and vertical axis.

5-queens Symmetry Example $\mathcal{S}$ = 180 Rotation



- Rotate by 180 degrees.
- x_1 exchanges with x_5 and x_2 with x_4.
- New domains $\theta(val) = 6 - val$ for each x_i.
- Equivalence classes:
  - Variables $\{x_1, x_2\}$, $\{x_2, x_4\}$ and $\{x_3\}$.
  - Values $\{1, 5\}$, $\{2, 4\}$, $\{3\}$.
- Reflection about the horizontal axis and vertical axis.
- Rotation by 360? Rotation by 90?

5-queens - Different Formulation

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

- $X = \{x_1, x_2, x_3, x_4, x_5\}$
- $D = \{1, 2, \ldots, 25\}$
- What are the symmetries here? Do they include domains, variables or both?

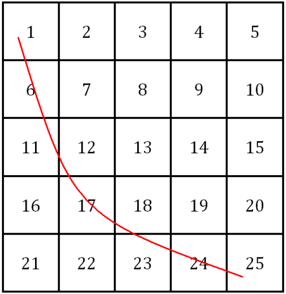
5-queens - Different Formulation

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

1	2	11	16	21
6	7	12	17	22
3	8	13	18	23
4	9	14	19	24
5	10	15	20	25

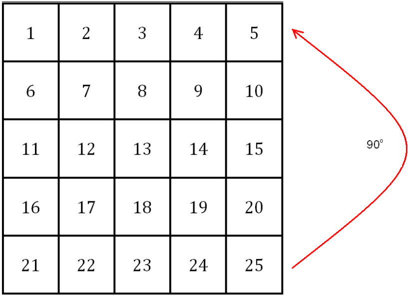
- $X = \{x_1, x_2, x_3, x_4, x_5\}$
- $D = \{1, 2, \ldots, 25\}$
- What are the symmetries here? Do they include domains, variables or both?

5-queens - Different Formulation



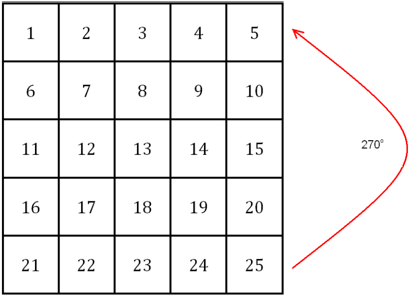
- $X = \{x_1, x_2, x_3, x_4, x_5\}$
- $D = \{1, 2, \ldots, 25\}$
- What are the symmetries here? Do they include domains, variables or both?

5-queens - Different Formulation



- $X = \{x_1, x_2, x_3, x_4, x_5\}$
- $D = \{1, 2, \ldots, 25\}$
- What are the symmetries here? Do they include domains, variables or both?

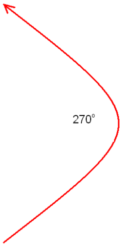
5-queens - Different Formulation



- $X = \{x_1, x_2, x_3, x_4, x_5\}$
- $D = \{1, 2, \ldots, 25\}$
- What are the symmetries here? Do they include domains, variables or both?

5-queens - Different Formulation

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25



- $X = \{x_1, x_2, x_3, x_4, x_5\}$
- $D = \{1, 2, \ldots, 25\}$
- What are the symmetries here? Do they include domains, variables or both?
- All 8 symmetries.

Formulation of CSP has Symmetry and not the Problem

- The definition of the symmetry applies to the definition of CSP and not to the problem itself.
- Different CSP's formulations of the same problem can have different symmetries.
- What symmetry to select?

Formulation of CSP has Symmetry and not the Problem

- The definition of the symmetry applies to the definition of CSP and not to the problem itself.
- Different CSP's formulations of the same problem can have different symmetries.
- What symmetry to select? What about one that produces the smallest number of equivalent classes?

Contents

Three Approaches for Symmetrical CSPs

Adding symmetry breaking global constraints

- Adding global constraints to convert it to an asymmetrical CSP.

Modify search

- Pruning symmetric states as they appear in search.

Modify search heuristics

- Using symmetry-breaking rules to guide search.

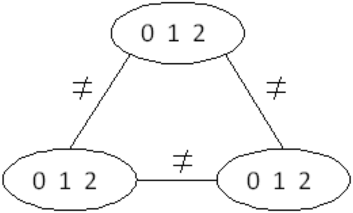
Removing Symmetry from the Problem - Global Symmetry

- Puget [93] while developing PECOS tool.
- Symmetry can cause a combinatorial explosion of the search space.
- Arc-consistency *AC* is not adapted to symmetrical CSPs. Ex. *Pigeon Hole* problem.
- In symmetrical CSP a *permutation of the variables* map one solution onto another solution.
- Removing symmetrical solutions by adding a constraint - if $C \subset C'$ then $Sol(P') \subset Sol(P)$ - reduction.
- Add static symmetry breaking constraints - *an ordering constraint* $x_1 < x_2 < \cdots < x_n$ - and do AC after that.

Creating a Global Constraint

Example

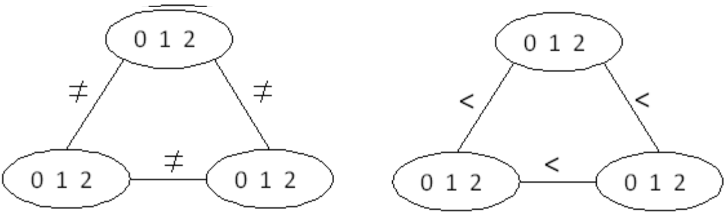
- $V = \{v_0, v_1, v_2\}$, $D = \{0, 1, 2\}$
- $C : v_0 \neq v_1 \wedge v_1 \neq v_2 \wedge v_2 \neq v_0$
- How many solutions?



Creating a Global Constraint

Example

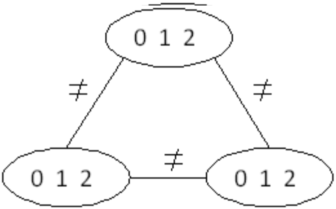
- $V = \{v_0, v_1, v_2\}$, $D = \{0, 1, 2\}$
- $C : v_0 \neq v_1 \wedge v_1 \neq v_2 \wedge v_2 \neq v_0$
- How many solutions?
- Has a symmetry (permutation): $v_0 \rightarrow v_1$, $v_1 \rightarrow v_2$, $v_2 \rightarrow v_0$

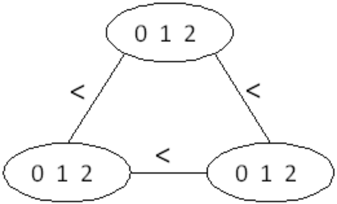


Creating a Global Constraint

Example

- $V = \{v_0, v_1, v_2\}$, $D = \{0, 1, 2\}$
- $C : v_0 \neq v_1 \wedge v_1 \neq v_2 \wedge v_2 \neq v_0$
- How many solutions?
- Has a symmetry (permutation): $v_0 \rightarrow v_1$, $v_1 \rightarrow v_2$, $v_2 \rightarrow v_0$
- Adding $v_0 < v_1 < v_2$ - How many solutions?





General Direction

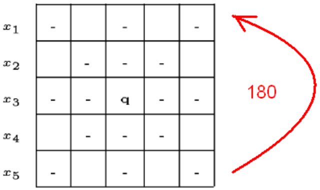
- Enforcing *GAC* on this global constraint reduces the problem.
- Depending on the decomposition of a problem *GAC* propagation can be *NPC*.
- In "other" constraint paper by Law at al. [CP07].
  - Proposed SIGLEX global constraint.
  - Its *GAC* propagation is *P*.
  - But it prunes only some symmetric values in general cases.

Symmetry is Dynamic [Meseguer & Torras 2001]

x_1	-		-		-
x_2		-	-	-	
x_3	-	-	q	-	-
x_4		-	-	-	
x_5	-		-		-

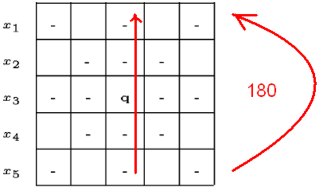
- Symmetries holding at the initial states is a global symmetry.

Symmetry is Dynamic [Meseguer & Torras 2001]



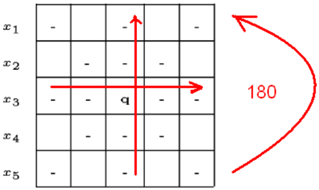
- Symmetries holding at the initial states is a global symmetry.

Symmetry is Dynamic [Meseguer & Torras 2001]



- Symmetries holding at the initial states is a global symmetry.

Symmetry is Dynamic [Meseguer & Torras 2001]



- Symmetries holding at the initial states is a global symmetry.

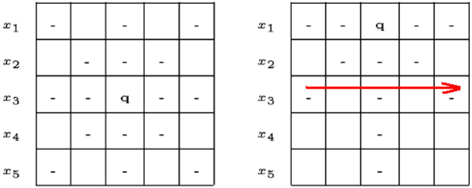
Symmetry is Dynamic [Meseguer & Torras 2001]

x_1	-		-		-
x_2		-	-	-	
x_3	-	-	q	-	-
x_4		-	-	-	
x_5	-		-		-

x_1	-	-	q	-	-
x_2		-	-	-	
x_3	-		-		-
x_4			-		
x_5			-		

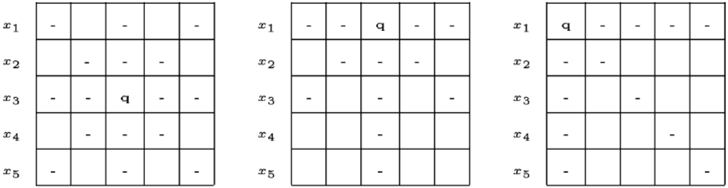
- Symmetries holding at the initial states is a global symmetry.
- After an assignment to v_i the global symmetry may break.

Symmetry is Dynamic [Meseguer & Torras 2001]



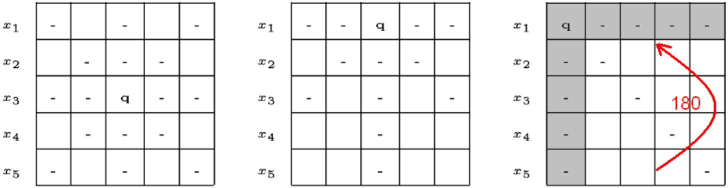
- Symmetries holding at the initial states is a global symmetry.
- After an assignment to v_i the global symmetry may break.

Symmetry is Dynamic [Meseguer & Torras 2001]



- Symmetries holding at the initial states is a global symmetry.
- After an assignment to v_i the global symmetry may break.

Symmetry is Dynamic [Meseguer & Torras 2001]



- Symmetries holding at the initial states is a global symmetry.
- After an assignment to v_i the global symmetry may break.
- Yet, new symmetries can appear in some states.

Symmetry is Dynamic

x_1	-	q	-	-	-
x_2	-	-	-		
x_3		-		-	
x_4		-			-
x_5		-			

- Symmetries holding at the initial states is a global symmetry.
- After an v_i assignment the global symmetry can break.
- Yet, new symmetries can appear in some states.
- Symmetries can be broken and restored during the search.

Symmetry is Dynamic

x_1	-	q	-	-	-
x_2	-	-	-		
x_3		-		-	
x_4		-			-
x_5		-			

x_1	-	q	-	-	-
x_2	-	-	-	-	
x_3		-		-	
x_4		-	-	-	-
x_5	-	-	-	q	-

- Symmetries holding at the initial states is a global symmetry.
- After an v_i assignment the global symmetry can break.
- Yet, new symmetries can appear in some states.
- Symmetries can be broken and restored during the search.

Pruning Symmetric States from Search

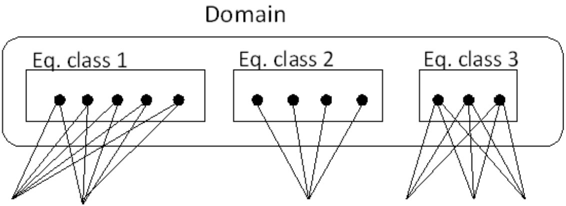
Symmetric Variables [Brown et al. 1989]

- Does not select *vvp* if *vvp* leads to a redundant partial assignment.
- Determines if a current partial assignment X is equivalent to a smaller assignment under a symmetry group G.
- Has pseudo code of the Backtracking Algorithm with Symmetries.
- Symmetries are given.

Pruning Symmetric States from Search

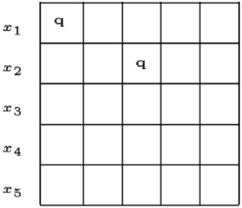
Symmetric Values [Freuder 1991]

- Only selects one *val* from equivalence class of values during *vvp* selection.
- Values *a* and *b* are neighborhood interchangeable if each *vvp* is consistent with their neighborhood.
- Algorithm to determine local value interchangeability is $O(n^2 d^2)$.
- Symmetries are discovered.



Symmetric Variables and Values [Backofen & Will CP99, Gent & Smith 2000]

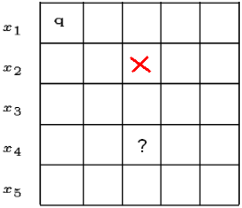
- Does not interfere with the heuristic searches (variable ordering).
- Adds symmetry breaking constraints to the right branches of search tree.



$x_1 = 2$, $x_2 = 3$ - backtracking

Symmetric Variables and Values [Backofen & Will CP99, Gent & Smith 2000]

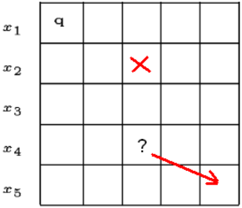
- Does not interfere with the heuristic searches (variable ordering).
- Adds symmetry breaking constraints to the right branches of search tree.



$x_1 = 2$, $x_2 = 3$ - backtracking
$x_1 = 2$, $x_2 \neq 3$ - should we consider $x_4 = 3$?

Symmetric Variables and Values [Backofen & Will CP99, Gent & Smith 2000]

- Does not interfere with the heuristic searches (variable ordering).
- Adds symmetry breaking constraints to the right branches of search tree.



$x_1 = 2$, $x_2 = 3$ - backtracking
$x_1 = 2$, $x_2 \neq 3$ - should we consider $x_4 = 3$? Depends if $x_5 = 5$ or not
If $x_5 \neq 5$ then $x_2 = 3$ and $x_3 = 3$ are not equivalent. Generally it is not known if $x_5 = 5$ or $x_5 \neq 5$.
Adding a conditional constraint $x_1 = 1 \wedge x_2 \neq 3 \wedge x_5 = 5 \Rightarrow x_4 \neq 3$.

Use Symmetry to Guide Search

Dynamic Variable Ordering [Meseguer & Torras 2001]

- Direct search toward subspaces with many non-symmetric states.
- Selecting *vvp* that breaks the most of the symmetries.
- It will lead to more evenly distributed solutions in the CSP's state space.
- More about it in my project presentation.

Contents

- Avoiding symmetric path in search [Glaischer 1874, Brown et al. 1989]
- Value interchangeability [Freuder 1991]
- Symmetry breaking constraints [Puget 93, Backofen & Will 99]
- Discovering symmetries
  - Equivalent to graph isomorphism.
  - Complexity unknown (P? NPC?)
  - Discover symmetry generators with Nauty, Saucy, AUTOM

Bibliography I

Cynthia A. Brown, Larry Finkelstein, and Paul Walton Purdom, Jr.
Backtrack searching in the presence of symmetry.
In *AAECC-6: Proceedings of the 6th International Conference, on Applied Algebra, Algebraic Algorithms and Error-Correcting Codes*, pages 99–110, London, UK, 1989. Springer-Verlag.

Rolf Backofen and Sebastian Will.
Excluding symmetries in constraint-based search.
In *CP '99: Proceedings of the 5th International Conference on Principles and Practice of Constraint Programming*, pages 73–87, London, UK, 1999. Springer-Verlag.

Darga et al.
Saucy.
http://vlsicad.eecs.umich.edu/BK/SAUCY.

Bibliography II

Eugene C. Freuder.
Eliminating interchangeable values in constraint satisfaction problems.
In *In Proceedings of AAAI-91*, pages 227–233, 1991.

J.W.L. Glaisher.
On the Problem of the Eight Queens.
Philosophical Magazine, 48:457–467, 1874.

Ian P. Gent and Barbara M. Smith.
Symmetry breaking in constraint programming.
In *Proceedings of ECAI-2000*, pages 599–603. IOS Press, 2000.

Y. C. Law, J. H. M. Lee, Toby Walsh, and J. Y. K. Yip.
Breaking symmetry of interchangeable variables and values.
In *Principles and Practice of Constraint Programming CP 2007*, pages 423–437, London, UK, 2007. Springer-Verlag.

Bibliography III

Brendan MacKay.
Nauty.
http://cs.anu.edu.au/people/bdm/nauty.

Pedro Meseguer and Carme Torras.
Exploiting symmetries within constraint satisfaction search.
Artif. Intell., 129(1-2):133–163, 2001.

Jean-Francois Puget.
On the satisfiability of symmetrical constrained satisfaction problems.
In *ISMIS '93: Proceedings of the 7th International Symposium on Methodologies for Intelligent Systems*, pages 350–361, London, UK, 1993. Springer-Verlag.

Jean-Francois Puget.
Automatic Detection of Variable and Value Symmetries, 2005.