Efficient Structural Symmetry Breaking

Efficient Structural Symmetry Breaking

Or: some work that We've been doing in the past 4 or 5 years.

Efficient Structural Symmetry Breaking

Or: some work that We've been doing in the past 4 or 5 years.

Joint work (only one name appears on all papers) between:

- Pierre Flener(Uppsala/Sabnci), Pascal Van Hentenryck (Brown),
- Justin Pearson (Uppsala), Meinolf Sellmann (Brown),
- and Magnus Ågren (Uppsala)

Some common ideas

- Try to understand the boundary between tractable and intractable symmetry breaking.

Some common ideas

- Try to understand the boundary between tractable and intractable symmetry breaking.
- Specialised search procedures to break symmetry,

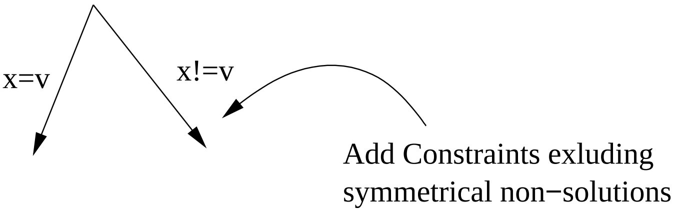
Some common ideas

- Try to understand the boundary between tractable and intractable symmetry breaking.
- Specialised search procedures to break symmetry,
- Better $O(n^{??})$ algorithms for symmetry breaking via matching rather than general computational group theory.

Symmetry Breaking

There are two (or three) techniques for breaking symmetry:

- Add constraints before search
- Dynamically break the symmetry during search



Types of Symmetry

- A CSP is a triple $(V, D, \{R_i(S_i)\}_{i \in I})$, V variables, D the domain of all variables, and a set of constraints.

Types of Symmetry

- A CSP is a triple $(V, D, \{R_i(S_i)\}_{i \in I})$, V variables, D the domain of all variables, and a set of constraints.
- A solution is a map $f : V \to D$ that satisfies all the constraints.
- Intuitive idea symmetries preserve solutions. A symmetry group, G, of a CSP acts on the set of maps $\mathrm{Hom}(V, D)$ that preserves solution

$$f \in \mathrm{Sol}(P) \wedge \pi \in G \text{ implies } \pi(f) \in \mathrm{Sol}(P)$$

Types of Symmetry

- We are interested in certain combinations of variable and value symmetries.

Types of Symmetry

- We are interested in certain combinations of variable and value symmetries.
  - A variable symmetry is a bijection $\sigma : V \rightarrow V$ such that $f : V \rightarrow D$ is a solution implies that $f \circ \sigma$ is a solution.

Types of Symmetry

- We are interested in certain combinations of variable and value symmetries.
  - A variable symmetry is a bijection $\sigma : V \to V$ such that $f : V \to D$ is a solution implies that $f \circ \sigma$ is a solution.
  - A value symmetry is a bijection $\pi : D \to D$ such that $f : V \to D$ is a solution implies that $\pi \circ f$ is a solution.

Types of Symmetry

- We are interested in certain combinations of variable and value symmetries.
  - A variable symmetry is a bijection $\sigma : V \to V$ such that $f : V \to D$ is a solution implies that $f \circ \sigma$ is a solution.
  - A value symmetry is a bijection $\pi : D \to D$ such that $f : V \to D$ is a solution implies that $\pi \circ f$ is a solution.
- Combinations f is a solution implies $\pi \circ f \circ \sigma$ is a solution.

Types of Symmetry Groups

We have to start somewhere:

- the symmetric group, consider CSPs where either or both π and σ come from the full symmetric group S_V or S_D.

Types of Symmetry Groups

We have to start somewhere:

- the symmetric group, consider CSPs where either or both π and σ come from the full symmetric group S_V or S_D. (sometimes called full symmetries)

Types of Symmetry Groups

We have to start somewhere:

- the symmetric group, consider CSPs where either or both π and σ come from the full symmetric group S_V or S_D. (sometimes called full symmetries)
- Partial symmetries. Given a partition $P = P_1 \cup \cdots \cup P_n$ study symmetries the preserve the partition $x \in P_i$ implies $x.\pi$ in P_i.

Types of Symmetry Groups

We have to start somewhere:

- the symmetric group, consider CSPs where either or both π and σ come from the full symmetric group S_V or S_D. (sometimes called full symmetries)
- Partial symmetries. Given a partition $P = P_1 \cup \cdots \cup P_n$ study symmetries the preserve the partition $x \in P_i$ implies $x.\pi$ in P_i.
- Various groups of wreath products. In the social golfer you can either permute whole weeks or within each week you can permute the groups.

The idea is that these symmetries are easy to understand and capture some interesting problems.

Wreath Products – Social Golfer

- Derive a n-week for p golfer divided into groups of size n with the constraints:
  - Each golfer plays each week

Wreath Products – Social Golfer

- Derive a n-week for p golfer divided into groups of size n with the constraints:
  - Each golfer plays each week
  - Each pair of golfer play in the same group at most once.

Wreath Products – Social Golfer

- Derive a n-week for p golfer divided into groups of size n with the constraints:
  - Each golfer plays each week
  - Each pair of golfer play in the same group at most once.
- Example, 3 weeks 9 golfer groups of size 3

Week 1	127	348	569
Week 2	135	249	678
Week 3	236	457	189

You can permute the weeks and within the weeks, but you can't take any two groups and swap them. This is exactly a wreath product.

Intractability

- A set valued CSP, $(V, D = 2^U, \{R_i(S_i)\})$ is where variables can be assigned subsets of a domain not just single elements.

Intractability

- A set valued CSP, $(V, D = 2^U, \{R_i(S_i)\})$ is where variables can be assigned subsets of a domain not just single elements.
- A set valued CSP is fully variable and value interchangeable if given $\pi \in S_V$ and $\sigma \in S_U$ and whenever f is a solution then

$$2^\sigma \circ f \circ \pi$$

is a solution.

Intractability

- A set valued CSP, $(V, D = 2^U, \{R_i(S_i)\})$ is where variables can be assigned subsets of a domain not just single elements.
- A set valued CSP is fully variable and value interchangeable if given $\pi \in S_V$ and $\sigma \in S_U$ and whenever f is a solution then

$$2^\sigma \circ f \circ \pi$$

  is a solution.

The dominance checking problem for such CSPs is NP-hard. The proof is not as easy as it first might seem. You have to show the NP-hardness for dominance detecting problems that arise during search.

Tractability

- One strand of research involves specialising the SBDX and seeing what you get.

Tractability

- One strand of research involves specialising the SBDX and seeing what you get.
- With full value symmetry the answer is quite simple.

Tractability

- One strand of research involves specialising the SBDX and seeing what you get.
- With full value symmetry the answer is quite simple.
- You only have to symmetry check against siblings.

Tractability

- One strand of research involves specialising the SBDX and seeing what you get.
- With full value symmetry the answer is quite simple.
- You only have to symmetry check against siblings.
- With full value symmetry the dominance check is very simple.

Tractability

- One strand of research involves specialising the SBDX and seeing what you get.
- With full value symmetry the answer is quite simple.
- You only have to symmetry check against siblings.
- With full value symmetry the dominance check is very simple.
  - At each node if you have used the values $d_1, \ldots, d_k$ then the equivalence classes are:

$$\{d_1\}, \ldots, \{d_k\}, \{d_{k+1}, \ldots, \}$$

  - Pick an previously used value or one new value.

Labelling Algorithm

```
bool fValllabel(𝒫) {
    return fValllabelA(𝒫,ϵ);
}
bool fValllabelA(⟨V, D, C⟩,θ) {
    if scope(θ) = V then
        return C(θ);
    select v in V \ scope(θ);
    A := image(θ);
    if A ≠ D then
        select f in D \ A; A := A ∪ {f};
    forall(d ∈ A)
        θ′ := θ & v = d;
        if ¬Failure(⟨V, D, C⟩, θ′) then
            if fValllabelA(⟨V, D, C⟩,θ′) then
                return true;
    return false;
}
```

Value Symmetry: Modified Search Procedures

- Generalises to wreath problems

Value Symmetry: Modified Search Procedures

- Generalises to wreath problems
- Generalised into a much more general context in GE-trees.

Value Symmetry: Modified Search Procedures

- Generalises to wreath problems
- Generalised into a much more general context in GE-trees.
- Although specialised search procedures often have better complexity bounds.

More Tractability: Piecewise value and Piecewise variable interchangeable

- The basic idea is to turn the dominance check into a matching problem.

More Tractability: Piecewise value and Piecewise variable interchangeable

- The basic idea is to turn the dominance check into a matching problem.
- Key idea build signatures the count how many times values appears in variable partitions.

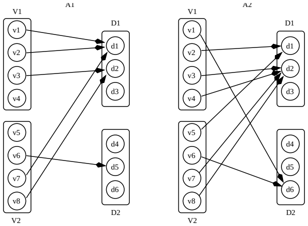
More Tractability: Piecewise value and Piecewise variable interchangeable

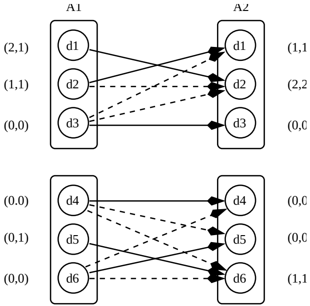
- The basic idea is to turn the dominance check into a matching problem.
- Key idea build signatures the count how many times values appears in variable partitions.
- Define $sig(d_i)$ to be a vector of values of how many times d_i appears in each variable partition.

More Tractability: Piecewise value and Piecewise variable interchangeable

- The basic idea is to turn the dominance check into a matching problem.
- Key idea build signatures the count how many times values appears in variable partitions.
- Define $sig(d_i)$ to be a vector of values of how many times d_i appears in each variable partition.
- Build a bipartite graph the has a perfect matching iff one assignments dominates the other.

Value Symmetry: Modified Search Procedures





Things I have no time to talk about

- Relationship between static symmetry breaking constraints and dynamic symmetry breaking.
- Depending on what levels of consistency you can make sensible comparisons dynamic search trees.

Constraints for Structural Symmetry Breaking

- Schedule study groups for two sets of five indistinguishable students each to sit at six indistinguishable tables (with four seats each).
- The variables $\{s_1, \ldots, s_5\} \cup \{s_6, \ldots, s_{10}\}$ correspond to the students.
- These variables are to be assigned to table values from the domain $\{t_1, \ldots, t_6\}$.

Structural Symmetry Breaking Constraints

- Order the variables within each class.
- Count how many times a value appears in each variable class.
- A signature for value is then a list of how many times each value appears in each class.
- Lex order the classes

Constraints for Structural Symmetry Breaking

- The static structural symmetry-breaking constraints are:

$$
\begin{array}{l}
s_1 \le s_2 \le s_3 \le s_4 \le s_5, \quad s_6 \le s_7 \le s_8 \le s_9 \le s_{10}, \\
\mathrm{gcc}(s_1, \ldots, s_5,\ t_1, \ldots, t_6,\ f_1^1, \ldots, f_6^1), \\
\mathrm{gcc}(s_6, \ldots, s_{10},\ t_1, \ldots, t_6,\ f_1^2, \ldots, f_6^2), \\
(f_1^1, f_1^2) \ge_{lex} (f_2^1, f_2^2) \ge_{lex} \cdots \ge_{lex} (f_6^1, f_6^2)
\end{array}
$$

- Consider the assignment

$$\{s_1 \mapsto t_1, s_2 \mapsto t_1, s_3 \mapsto t_2, s_4 \mapsto t_2, s_5 \mapsto t_3$$
$$s_6 \mapsto t_1, s_7 \mapsto t_2, s_8 \mapsto t_3, s_9 \mapsto t_4, s_{10} \mapsto t_5\}.$$

- Observe that the $\geq_{lex}$ constraints are satisfied, because

$$(2,1) \geq_{lex} (2,1) \geq_{lex} (1,1) \geq_{lex} (0,1) \geq_{lex} (0,1) \geq_{lex} (0,0).$$

A Student Moves

- If student 10 moves from table 5 to table 6, producing a symmetrically equivalent assignment because the tables are fully interchangeable:

$$\{s_1 \mapsto t_1, s_2 \mapsto t_1, s_3 \mapsto t_2, s_4 \mapsto t_2, s_5 \mapsto t_3$$
$$s_6 \mapsto t_1, s_7 \mapsto t_2, s_8 \mapsto t_3, s_9 \mapsto t_4, s_{10} \mapsto t_6\}$$

- then the $\geq_{lex}$ constraints are no longer satisfied, because

$$(2, 1) \geq_{lex} (2, 1) \geq_{lex} (1, 1) \geq_{lex} (0, 1) \geq_{lex} (0, 0) \not\geq_{lex} (0, 1).$$

Questions

- structural info can we get from symmetry groups?
- Can we do better with the tractable/intractable boundary rather the just proving special cases?
- How do structural properties of groups relate to the computational complexity of symmetry breaking?
- Is there some nice notion of closure of sets of symmetry groups that preserves complexity?

All of the papers can possibly be found at `http://www.it.uu.se/research/astra` or more likely by following the links from `http://user.it.uu.se/~justin`.