{Symmetry, Logic, CSP}

Libor Barto

Charles University in Prague

{Symmetry, Logic, Computation}
Simons Institute, Berkeley, 9 Nov 2016

Message

Topic: Constraint Satisfaction Problem (CSP)
over a fixed finite template
– **a class of computational problems**

In this context

- A problem is hard $\Leftrightarrow$ it lacks symmetry
  - lacks symmetry $\Rightarrow$ can simulate many problems $\Rightarrow$ hard
    1 reason for hardness
  - symmetry can be exploited in algorithms (directly/indirectly)
    1 (?) algorithm scheme for all easy cases
- The most popular symmetries (eg. automorphisms) are useless

In general

- Goes beyond this particular class
- How far? **Still a big hole in the market**

Topic: CSP over a fixed finite template

In this context

- A problem is hard $\Leftrightarrow$ it lacks symmetry
  - lacks symmetry $\Rightarrow$ can simulate many problems $\Rightarrow$ hard
    1 reason for hardness
  - symmetry can be exploited in algorithms (directly/indirectly)
    1 (?) algorithm scheme for all easy cases
- The most popular symmetries (eg. automorphisms) are useless

In general

- Goes beyond this particular class
- How far? **Still a big hole in the market**

CSP over fixed finite template

Fix $\mathbb{A} = (A; R_1, R_2, \ldots, R_n)$: finite relational structure
each $R_i \subset A^k$ or $R_i : A^k \to \{\texttt{true}, \texttt{false}\}$

Definition

Instance of $\mathrm{CSP}(\mathbb{A})$: primitive positive sentence, eg.

$(\exists x)(\exists y)(\exists z)(\exists t)\ R_1(x, y, z) \wedge R_2(t, z) \wedge R_1(y, y, z)$

where each R_i is in $\mathbb{A}$.

Question: Is it true?

- **Other variants**: infinite A; nothing is fixed; something else is fixed; different connectives
- **Other questions**
  - Count the number of solutions
  - Optimize the number of satisfied constraints
  - Approximately optimize the number of satisfied constraints

Examples and a conjecture

- 2-SAT: $\mathbb{A} = (\{0,1\}; x \vee y, x \vee \neg y, \neg x \vee \neg y)$
- 3-SAT: $\mathbb{A} = (\{0,1\}; x \vee y \vee z, x \vee y \vee \neg z, \dots)$
- HORN-3-SAT: $\mathbb{A} = (\{0,1\}; x = 0, x = 1, x \wedge y \to z)$

- Directed st–connectivity: $\mathbb{A} = (\{0,1\}; x = 0, x = 1, x \leq y)$
- Undirected st–connectivity: $\mathbb{A} = (\{0,1\}; x = 0, x = 1, x = y)$

- 3-COLOR: $\mathbb{A} = (\{0,1,2\}; x \neq y)$
- p-3-LIN: $\mathbb{A} = (GF(p); x + y + z = 0, x + 2y + 3z = 10, \dots)$

Conjecture (The dichotomy conjecture [Feder and Vardi'93])

For every $\mathbb{A}$, $\mathrm{CSP}(\mathbb{A})$ is either in P or NP-complete.

Selected results

- **The dichotomy conjecture is true**:
  - if $|A| = 2$ [Schaefer'78]
  - if $\mathbb{A} = (A; R)$, R is binary and symmetric [Hell and Nešetřil'90]
  - if $|A| = 3$ [Bulatov'06]
  - if $\mathbb{A}$ contains all unary relations [Bulatov'03 '16] [Barto'11]
  - if $\mathbb{A} = (A; R)$ where R is binary, without sources or sinks
    [Barto, Kozik, Niven'09]
  - in general? [Zhuk?]
- **Applicability of known algorithmic principles understood**:
  - Describing all solutions
    [Idziak, Markovic, McKenzie, Valeriote, Willard'07]
  - Local consistency (constraint propagation)
    [Barto, Kozik'09], [Bulatov]
  - All known tractable cases solvable by a combination of these two
- **Work on finer complexity classification**

Topic: CSP over a fixed finite template

In this context

- A problem is hard $\Leftrightarrow$ it lacks symmetry
  - lacks symmetry $\Rightarrow$ can simulate many problems $\Rightarrow$ hard
    1 reason for hardness
  - symmetry can be exploited in algorithms (directly/indirectly)
    1 (?) algorithm scheme for all easy cases
- The most popular symmetries (eg. automorphisms) are useless

In general

- Goes beyond this particular class
- How far? **Still a big hole in the market**

Example of simulation (gadget reduction, pp-definition)

- $\mathbb{A} = (A; R)$, where R is ternary
- $\mathbb{B} = (A; S, T)$, where S is binary and T is unary
  - $S(x, y)$ iff $(\exists z)\ R(x, y, z) \wedge R(y, y, x)$
  - $T(x)$ iff $R(x, x, x)$

- Each instance of $\mathrm{CSP}(\mathbb{B})$, eg.

$$(\exists x)(\exists y)(\exists z)\ T(z) \wedge S(x, y)$$

- can be rewritten to an equivalent instance of $\mathrm{CSP}(\mathbb{A})$

$$(\exists x)(\exists y)(\exists z)(\exists z')\ R(z, z, z) \wedge R(x, y, z') \wedge R(y, y, x)$$

- Thus $\mathrm{CSP}(\mathbb{B})$ is easier than $\mathrm{CSP}(\mathbb{B})$

1 reason for hardness

- **Fact:** If
  - $\mathbb{A}$ pp-defines $\mathbb{B}$

    definition like in the previous slide
  - or more generally, $\mathbb{A}$ pp-interprets $\mathbb{B}$

    powers allowed $\leftrightarrow$ variables encoded by tuples of variables
  - or more generally, $\mathbb{A}$ pp-constructs $\mathbb{B}$

    homomorphic equivalence allowed

  then $\mathrm{CSP}(\mathbb{B})$ is easier than $\mathrm{CSP}(\mathbb{A})$
- **Corollary:** If $\mathbb{A}$ pp-constructs some structure with NP-hard CSP (like 3–SAT), then $\mathrm{CSP}(\mathbb{A})$ is NP-hard
- **Remark:** $\mathbb{A}$ pp-constructs 3–SAT $\Rightarrow$ $\mathbb{A}$ pp-constructs every finite structure
- **Tractability conjecture:** If $\mathbb{A}$ does not pp-construct 3–SAT then $\mathrm{CSP}(\mathbb{A})$ is in P

  [Feder,Vardi'93] [Bulatov,Jeavons,Krokhin'00] [Bodirsky] [Willard]

Digression: Group theory vs. Universal algebra

Too popular viewpoint

Group theory, Semigroup theory

- group: algebraic structure $\mathbf{G} = (G; \cdot, ^{-1}, 1)$ satisfying ...
- permutation group: when G happens to be a set of bijections, $\cdot$ is composition, ...
- monoid: algebraic structure $\mathbf{M} = (M; \cdot, 1)$ satisfying ...
- transformation monoid: ...

Universal algebra

- algebra: any algebraic structure $\mathbf{Z} = (Z;\ \text{some operations}\ )$

Rants

- Model theorist: models of purely algebraic signature, why do you avoid relations?
- Algebraist: groups are complicated enough, nothing interesting can be said about general algebras
- All: have you ever seen a 37-ary operation? You shouldn't study such a nonsense

Alternative viewpoint

	concrete	abstract
unary invert. symmetries	permutation group	group
unary symmetries	transformation monoid	monoid
higher arity symmetries	function clone	abstract clone

- permutation group: Subset of $\{f : A \to A\}$ closed under composition and id_A and inverses...

  can be given by a generating unary algebra

- group: Forget concrete mappings, remember composition
- function clone: Subset of $\{f : A^n \to A : n \in \mathbb{N}\}$ closed under composition and projections

  can be given by a generating algebra

- abstract clone: Forget concrete mappings, remember composition

  aka variety, finitary monad over SET, Lawvere theory

End of digression

Topic: CSP over a fixed finite template

In this context

- A problem is hard $\Leftrightarrow$ it lacks symmetry
  - lacks symmetry $\Rightarrow$ can simulate many problems $\Rightarrow$ hard
    1 reason for hardness
  - symmetry can be exploited in algorithms (directly/indirectly)
    1 (?) algorithm scheme for all easy cases
- The most popular symmetries (eg. automorphisms) are useless

In general

- Goes beyond this particular class
- How far? **Still a big hole in the market**

Polymorphisms

Objects capturing symmetry of $\mathrm{CSP}(\mathbb{A})$

- $\mathsf{Aut}(\mathbb{A}) = \{f : \mathbb{A} \to \mathbb{A}$ automorphism$\}$ automorphism group
- $\mathsf{End}(\mathbb{A}) = \{f : \mathbb{A} \to \mathbb{A}$ homomorph.$\}$ endomorphism monoid
- $\mathsf{Pol}(\mathbb{A}) = \{f : \mathbb{A}^n \to \mathbb{A}$ homomorphism$\}$ polymorphism clone

Trivial clone $\mathcal{T}$ – contains only projections

- aka 0,1,2
- **Example:** Pol(3–SAT)

Topic: CSP over a fixed finite template

In this context

- A problem is hard $\Leftrightarrow$ it lacks symmetry
  - lacks symmetry $\Rightarrow$ can simulate many problems $\Rightarrow$ hard
    1 reason for hardness
  - symmetry can be exploited in algorithms (directly/indirectly)
    1 (?) algorithm scheme for all easy cases
- The most popular symmetries (eg. automorphisms) are useless

In general

- Goes beyond this particular class
- How far? **Still a big hole in the market**

Expressive power and polymorphisms

Theorem ([Birkhoff'35] [Geiger'68] [Bodnarčuk et al.'69] [Bodirsky] [Willard] [Barto, Opršal, Pinsker])

- $\mathbb{A}$ *pp-defines* $\mathbb{B}$ *iff* $\mathrm{Pol}(\mathbb{A}) \subseteq \mathrm{Pol}(\mathbb{B})$
- $\mathbb{A}$ *pp-interprets* $\mathbb{B}$ *iff* $\mathrm{Pol}(\mathbb{A}) \to \mathrm{Pol}(\mathbb{B})$ *(homo)*
- $\mathbb{A}$ *pp-constructs* $\mathbb{B}$ *iff* $\mathrm{Pol}(\mathbb{A}) \dashrightarrow \mathrm{Pol}(\mathbb{B})$ *(h1 homo)*

Example: 3–SAT pp-interprets every structure

Remarks

- Proofs constructive $\Rightarrow$ generic reductions
- $f : \mathrm{Pol}(\mathbb{A}) \to \mathrm{Pol}(\mathbb{B})$ is a homo iff it preserves equations (eg. associative operation $\mapsto$ associative operation)
- $f : \mathrm{Pol}(\mathbb{A}) \dashrightarrow \mathrm{Pol}(\mathbb{B})$ is a h1 homo iff it preserves equations of height 1 (eg. commutative op. $\mapsto$ commutative op.)

Tractability conjecture again

Tractability conjecture

If $\nexists$ $\mathrm{Pol}(\mathbb{A}) \dashrightarrow \mathcal{T}$, *then* $\mathrm{CSP}(\mathbb{A})$ *in P.*

Recall: Otherwise $\mathrm{CSP}(\mathbb{A})$ is NP-complete.

Theorem

TFAE

- $\mathbb{A}$ *does not pp-construct all finite ie.* $\nexists$ *homo* $\mathrm{Pol}(\mathbb{A}) \dashrightarrow \mathcal{P}$
  ie. polymorphisms satisfy nontrivial equations
- $\mathrm{Pol}(\mathbb{A})$ *contains an operation* s *of arity* 4 *such that*
  $s(a, r, e, a) = s(r, a, r, e)$
  [Siggers'10], [Kearnes,Marković,McKenzie'14]
- $\mathrm{Pol}(\mathbb{A})$ *contains an operation* c *of arity* > 1 *such that*
  $c(a_1, a_2, \ldots, a_n) = c(a_2, \ldots, a_n, a_1)$ *[Barto,Kozik'12]*

3rd and 4th items: concrete and positive alternatives

Tractability conjecture vs. reality

Conjecture

TFAE (if $P \neq NP$)

- $\mathrm{CSP}(\mathbb{A})$ *is in P*
- $\mathbb{A}$ *has a polymorphism s such that* $s(a, r, e, a) = s(r, a, r, e)$

Even if the conjecture is wrong, we **know** that $\mathrm{CSP}(\mathbb{A})$ depends only on height 1 equations

Topic: CSP over a fixed finite template

In this context

- A problem is hard $\Leftrightarrow$ it lacks symmetry
  - lacks symmetry $\Rightarrow$ can simulate many problems $\Rightarrow$ hard
    1 reason for hardness
  - symmetry can be exploited in algorithms (directly/indirectly)
    1 (?) algorithm scheme for all easy cases
- The most popular symmetries (eg. automorphisms) are useless

In general

- Goes beyond this particular class
- How far? **Still a big hole in the market**

Endomorphism monoids are useless

- $\forall \mathbb{A} \quad \exists \mathbb{B}$ such that
  - $\mathbb{A}$ pp-constructs $\mathbb{B}$ pp-constructs $\mathbb{A}$ (ie. the same complexity)
  - $\mathrm{Aut}(\mathbb{B}) = \mathrm{End}(\mathbb{B}) = \{\mathrm{id}_B\}$

- $\forall \mathbb{A}, \mathbb{B}$ there is $\mathrm{End}(\mathbb{A}) \to \mathrm{End}(\mathbb{B})$

Topic: CSP over a fixed finite template

In this context

- A problem is hard $\Leftrightarrow$ it lacks symmetry
  - lacks symmetry $\Rightarrow$ can simulate many problems $\Rightarrow$ hard
    1 reason for hardness
  - symmetry can be exploited in algorithms (directly/indirectly)
    1 (?) algorithm scheme for all easy cases
- The most popular symmetries (eg. automorphisms) are useless

In general

- Goes beyond this particular class
- How far? **Still a big hole in the market**

UA in CSP

How universal algebra helps in CSP

- tools
- identifying intermediate cases

How polymorphisms are used

- Direct: A way to combine solutions to get another solution
- Indirect: Proving correctness

Two algorithmic ideas:

- Describe all solutions (direct)
- Refute unsolvable instances by enforcing consistency (indirect)

Describing all solutions

- Consider an instance of $\mathrm{CSP}(\mathbb{A})$ with n variables
- The set of solutions is $S \subseteq A^n$ invariant under $\mathrm{Pol}(\mathbb{A})$
- Can happen $|S| = |A|^n \Rightarrow$ cannot list all solutions
- **Idea: find a generating set of S, needs to be small**
- Example: $\mathrm{CSP}(\mathbb{A}) = p$–LIN
  - $\mathrm{Pol}(\mathbb{A}) =$ affine combinations
  - S is affine subspace of $GF(p)^n$
  - S has generating set of size $\leq (n+1)$
  - eg. A^2 generated by $(0,0),(0,1),(1,0)$
- UA $\Rightarrow$ obvious more general polymorphisms to look at
  Malcev [Bulatov'02], [Bulatov,Dalmau'06]
- UA $\Rightarrow$ another class where small generating sets exist
  Near unanimity [Baker,Pixley'75]
- UA $\Rightarrow$ class covering these two

Describing all solutions 2

Theorem (Berman, Idziak, Markovic, McKenzie, Valeriote, Willard'10)

TFAE

- *All invariant n-ary relations have small generating sets ($\leq$ polynomial in n)*
- *The number of n-ary invariant relations is small ($\leq$ exponential in n)*

In this case, $\mathrm{CSP}(\mathbb{A})$ *is in P. Moreover, a generating set of all solutions can be found in P–time.*

Local consistency

Roughly: $\mathbb{A}$ has bounded width iff $\mathrm{CSP}(\mathbb{A})$ can be solved by checking local consistency

More precisely:

- Fix $k \leq l$ (integers)
- (k, l)-algorithm: Derive the strongest constraints on k variables which can be deduced by "considering" l variables at a time.
- If a contradiction is found, answer "no"
  otherwise answer "yes"
- "no" answers are always correct
- if "yes" answers are correct for every instance of $\mathrm{CSP}(\mathbb{A})$ we say that $\mathbb{A}$ has width (k, l).
- if $\mathbb{A}$ has width (k, l) for some k, l then $\mathbb{A}$ has bounded width

Various equivalent formulations (bounded tree width duality, definability in Datalog, least fix point logic)

Local consistency 2

- $\mathbb{A}$ has a semilattice polymorphism $\Rightarrow \mathrm{CSP}(\mathbb{A})$ has width 1 [Feder,Vardi'93]
- $\mathbb{A}$ has a near unanimity polymorphism of arity $(n+1) \Rightarrow \mathrm{CSP}(\mathbb{A})$ has width n [Feder,Vardi'93]
- p-LIN does not have bounded width [Feder, Vardi'93]
- **Conjecture:** $\mathbb{A}$ has bounded width iff $\mathbb{A}$ does not pp-construct p–LIN [Larose, Zádori'07]
- UA suggests what to do next
  - 2-semilattices [Bulatov'06]
  - CD(3) [Kiss, Valeriote'07]
  - CD(4) [Carvalho, Dalmau, Marković, Maróti'09]
  - CD [Barto,Kozik'09]

Local consistency 3

Theorem

TFAE

1. $\mathbb{A}$ *does not pp-construct p-LIN*
2. $\mathbb{A}$ *has bounded width* [Barto, Kozik'09]
3. $\mathbb{A}$ *has width* $(2, 3)$ [Barto'16] [Bulatov]
4. $\mathrm{CSP}(\mathbb{A})$ *is decided by singleton arc consistency* [Kozik]
5. *the canonical semidefinite programming relaxation correctly decides* $\mathrm{CSP}(\mathbb{A})$ [Barto, Kozik'16]

Topic: CSP over a fixed finite template

In this context

- A problem is hard $\Leftrightarrow$ it lacks symmetry
  - lacks symmetry $\Rightarrow$ can simulate many problems $\Rightarrow$ hard
    1 reason for hardness
  - symmetry can be exploited in algorithms (directly/indirectly)
    1 (?) algorithm scheme for all easy cases
- The most popular symmetries (eg. automorphisms) are useless

In general

- Goes beyond this particular class
- How far? **Still a big hole in the market**

Beyond

- **Optimisation**
  - Complexity captured by weighted polymorphisms [Cohen, Cooper, Creed, Jeavons, Živný'13]
  - Even for valued CSP
  - Tractability conjecture ⇒ dichotomy for optimisation [Kolmogorov, Krokhin, Rolínek'15]
- **Exact counting # solutions**
  - Complexity captured by polymorphisms [Bulatov, Dalmau'03] [Bulatov,Grohe'05]
  - Dichotomy [Bulatov'08] [Dyer, Richerby'10]
- **Robust satisfiability** – almost solutions on almost satisfiable instances
  - Complexity captured by polymorphisms [Dalmau,Krokhin'11]
  - Dichotomy: in P if doesn't pp-construct p-LIN, otherwise NP-c [Hastad'01] [Barto,Kozik'12]

Beyond 2

Infinite domains

- All decision problems up to P–time reductions [Bodirsky,Grohe'08]
- Restrict to ω-categorical (aka oligomorphic)
  - Complexity captured by polymorphisms [Bodirsky,Nešetřil'06]
  - Actually abstract polymorphism clone + topology [Bodirsky,Pisnker'15]
  - Still "almost" covers all decision problems [Bodirsky,Grohe'08]
- Restrict even more
  - back to NP
  - P/NP-c dichotomy conjecture [Bodirsky,Pinsker'11], [Barto, Pinsker'16] [Barto, Opršal, Pinsker] [Olšák]

Topic: CSP over a fixed finite template

In this context

- A problem is hard $\Leftrightarrow$ it lacks symmetry
  - lacks symmetry $\Rightarrow$ can simulate many problems $\Rightarrow$ hard
    1 reason for hardness
  - symmetry can be exploited in algorithms (directly/indirectly)
    1 (?) algorithm scheme for all easy cases
- The most popular symmetries (eg. automorphisms) are useless

In general

- Goes beyond this particular class
- How far? **Still a big hole in the market**

How far?

Optimistic: everywhere

Realistic – ish

- Approximation
  - Complexity captured by approximate polymorphisms if UGC [Raghavendra'08]
  - Challenge: hardness part
- Hybrid CSPs (edge CSP, planar CSP, . . .)
  - eg. Perfect matching problem in graphs
  - What is the right notion of symmetry?
- Approximate hybrid, approximate counting, hybrid counting
  - eg. Holant problems
  - What is the right notion of symmetry?
- Infinite domain CSP
  - Explore the theory for larger classes (eg. to include linear programming)
  - Criterion for undecidability?

Big hole in the market

Do you see gadgets? Find symmetry!

Topic: CSP over a fixed finite template

In this context

- A problem is hard $\Leftrightarrow$ it lacks symmetry
  - lacks symmetry $\Rightarrow$ can simulate many problems $\Rightarrow$ hard
    1 reason for hardness
  - symmetry can be exploited in algorithms (directly/indirectly)
    1 (?) algorithm scheme for all easy cases
- The most popular symmetries (eg. automorphisms) are useless

In general

- Goes beyond this particular class
- How far? **Still a big hole in the market**

Thank you!