Constraint Satisfaction Problems
Part I: Complexity, Logic, Algebra

Libor Barto

Department of Algebra, Charles University, Prague

Caleidoscope, Paris, June 2019

CoCoSym: Symmetry in Computational Complexity

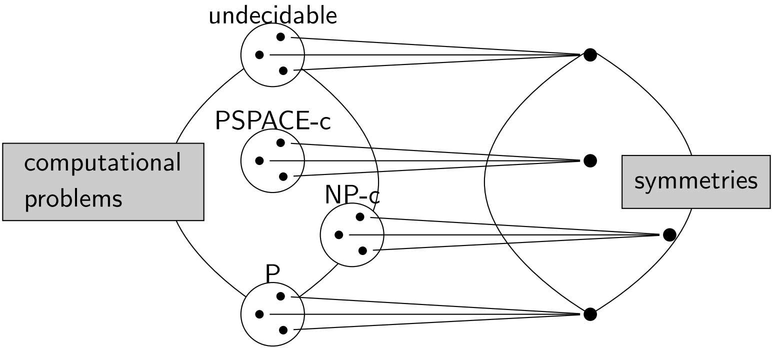
This project has received funding from the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme (grant agreement No 771005)

. . . there will also be three more focused topics, providing examples of (. . .) interactions between logic, algebra, and complexity

- Part I: logic, algebra, and complexity in **Constraint Satisfaction Problems (CSPs)** over fixed finite templates
- Part II: analysis, probability, and topology in (a variant of) CSP

In CSP: complexity captured by symmetry

$$\mathrm{CoolFunc}\colon \text{computational problems} \longrightarrow \text{objects capturing symmetry}$$
$$\text{kernel of } \mathrm{CoolFunc} \;=\; \text{polynomial time reducibility}$$



CSPs over fixed finite templates

- tiny portion of problems on the left
- kernel $\subsetneq$ polynomial time reducibility

CSP

Fix $\mathbb{A} = (A; R, S, \dots)$ relational structure

Definition ($\mathrm{CSP}(\mathbb{A})$)

Input: pp-sentence ϕ, eg. $(\exists x_1 \exists x_2 \dots) R(x_1, x_3) \wedge S(x_5, x_2) \wedge \dots$
Answer Yes: ϕ satisfied in $\mathbb{A}$
Answer No: ϕ not satisfied in $\mathbb{A}$

Search version: Find a satisfying assignment.
Search looks harder, but it's not [Bulatov, Jeavons, Krokhin'05]

Fact: Always in NP.

$\mathbb{K}_3 = (A; R)$ where

- $A = \{\textit{lilac}, \textit{mauve}, \textit{cyclamen}\}$
- $R =$ (binary) inequality relation on A

Input of $\mathrm{CSP}(\mathbb{K}_3)$ is, e.g.

$(\exists x_1 \exists x_2 \dots \exists x_4) R(x_1, x_2) \wedge R(x_1, x_3) \wedge R(x_1, x_4) \wedge R(x_2, x_3) \wedge R(x_2, x_4)$

Viewpoint

- variables = vertices
- clauses (constraints) = edges

$\mathrm{CSP}(\mathbb{K}_3)$ is the 3-coloring problem for graphs

Fact: It is NP-hard (7-coloring NP-hard, 2-coloring in P)

- $3\mathbb{NAE}_2 = (\{0,1\}; 3NAE_2)$ where
  $3NAE_2 = \text{all but } \{(0,0,0),(1,1,1)\}$
  $\mathrm{CSP}(3\mathbb{NAE}_2) =$ positive not-all-equal 3-SAT
  $=$ 2-coloring problem for 3-uniform hypergraphs

- $3\mathbb{NAE}_4 = (\{0,1,2,3\}; 3NAE_4)$, where $3NAE_4$ still ternary
  $\mathrm{CSP}(3\mathbb{NAE}_4) =$ 4-coloring problem for 3-uniform hypergraphs

- $1\mathbb{IN}3 = (\{0,1\}; 1in3)$ where
  $1in3 = \{(0,0,1),(0,1,0),(1,0,0)\}$
  $\mathrm{CSP}(1\mathbb{IN}3) =$ positive 1-in-3 SAT

Fact: All NP-hard

$3\mathbb{LIN}_5 = (\mathbb{Z}_5; L_{0000}, L_{0001}, \dots, L_{4444})$ where e.g.

$L_{1234} = \{(x, y, z) \in \mathbb{Z}_5^3 : 1x + 2y + 3z = 4\}$

(note: relations are affine subspaces of $\mathbb{Z}_5^3$)

$\mathrm{CSP}(3\mathbb{LIN}_5)$ = solving systems of linear equations in $\mathbb{Z}_5$

Fact: In P

Symmetry

polymorphism of $\mathbb{A}$: mapping $f : A^n \to A$
compatible with every relation

compatible with R: f applied component-wise to tuples in R
is a tuple in R

Example: $f(x_1, \ldots, x_4) = 2x_1 + 3x_2 + 3x_3 + 3x_4$ $\quad f : \mathbb{Z}_5^4 \to \mathbb{Z}_5$
is compatible with each L_{abcd}
because $f(\mathbf{v}_1, \ldots, \mathbf{v}_4)$ is an affine combination of these vectors (as $2 + 3 + 3 + 3 = 1$)
and L_{abcd} is an affine subspace

$\mathrm{Pol}(\mathbb{A})$: the set of all polymorphisms (it is a "clone")
$=$ set of (multivariable) symmetries of $\mathbb{A}$

Jeavons'98: On the algebraic structure of combinatorial problems

Theorem

Complexity of $\mathrm{CSP}(\mathbb{A})$ *is determined by* Pol($\mathbb{A}$)*:*
If Pol($\mathbb{A}$) $\subseteq$ Pol($\mathbb{B}$) *then* $\mathrm{CSP}(\mathbb{B})$ *reduces to* $\mathrm{CSP}(\mathbb{A})$*.*

Proof.

If Pol($\mathbb{A}$) $\subseteq$ Pol($\mathbb{B}$), then relations in $\mathbb{B}$ can be defined from relations in $\mathbb{A}$ by a pp-formula.

[Geiger'69, Bondarčuk, Kalužnin, Kotov, Romov'69]

This gives a computational reduction of $\mathrm{CSP}(\mathbb{B})$ to $\mathrm{CSP}(\mathbb{A})$. □

So: $\mathrm{CSP}(3\mathbb{LIN}_5)$ is in P because $3\mathbb{LIN}_5$ has a lot of polymorphs
$\mathrm{CSP}(1\mathbb{IN}3)$ is NP-complete because $1\mathbb{IN}3$ has few

System of functional equations is, e.g.

$$f(g(x,y),z) = g(x,h(y,z))$$
$$m(y,x,x) = m(y,y,y)$$
$$m(x,x,y) = m(y,y,y)$$

Satisfied in $\mathcal{M}$, where $\mathcal{M}$ is a set of functions:
symbols can be interpreted in $\mathcal{M}$ so that
each equality is (universally) satisfied

Example: The above system is satisfied in Pol($3\mathbb{LIN}_5$):

- take $f(x,y) = g(x,y) = h(x,y) = x$
  (note: projections are always polymoprhisms)
- take $m(x,y,z) = x - y + z$

Bulatov, Jeavons, Krokhin'05: Classifying the complexity of constraints using finite algebras + Bodirsky'08: PhD thesis

Theorem

Complexity of $\mathrm{CSP}(\mathbb{A})$ *is determined by*
systems of functional equations satisfied in Pol($\mathbb{A}$):
If each system satisfied in Pol($\mathbb{A}$) *is satisfied in* Pol($\mathbb{B}$),
then $\mathrm{CSP}(\mathbb{B})$ *reduces to* $\mathrm{CSP}(\mathbb{A})$.

Proof.

Previous theorem, pp-definitions $\rightarrow$ pp-interpretations,
the HSP theorem [Birkhoff'35] □

So: $\mathrm{CSP}(3\mathbb{LIN}_5)$ is in P because
Pol($3\mathbb{LIN}_5$) satisfies strong systems of functional equations.

Barto, Opršal, Pinsker'18: The wonderland of reflections

minor condition = system of functional equations, each of the form

symbol(*variables*) = *symbol*(*variables*),

e.g. $m(y,x,x) = m(y,y,y)$, $m(x,x,y) = m(y,y,y)$

Theorem

Complexity of $\mathrm{CSP}(\mathbb{A})$ *determined by*

minor conditions satisfied in $\mathrm{Pol}(\mathbb{A})$*:*

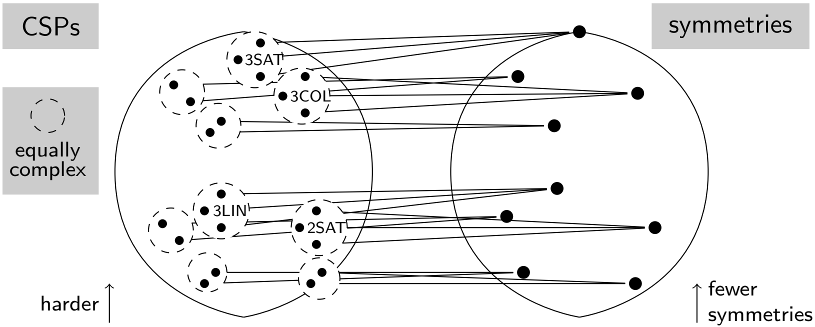
If each minor condition satisfied in $\mathrm{Pol}(\mathbb{A})$ *is satisfied in* $\mathrm{Pol}(\mathbb{B})$*,*

then $\mathrm{CSP}(\mathbb{B})$ *reduces to* $\mathrm{CSP}(\mathbb{A})$*.*

Proof.

pp-interpretation $\to$ pp-construction,

version of the HSP theorem. □

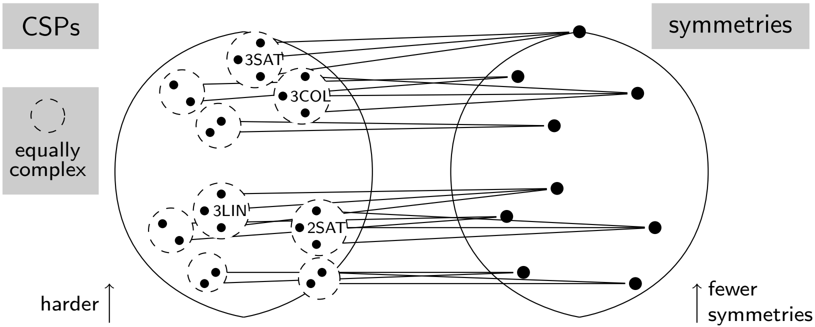


(1) polymorphisms

(2) systems of functional equations satisfied by polymorphisms

(3) minor conditions satisfied by polymorphisms

Where are the borderlines between complexity classes?

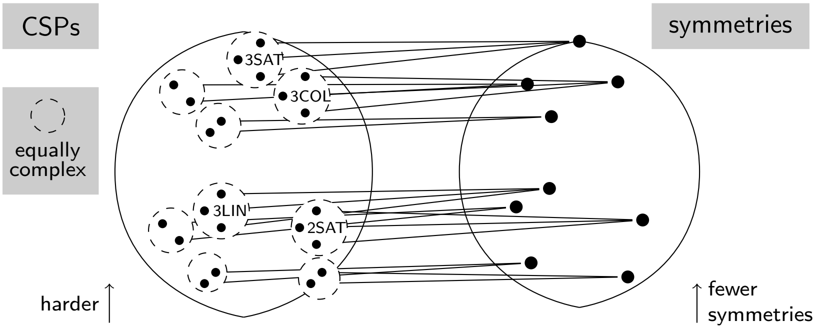


(1) polymorphisms

(2) systems of functional equations satisfied by polymorphisms

(3) minor conditions satisfied by polymorphisms

Where are the borderlines between complexity classes?

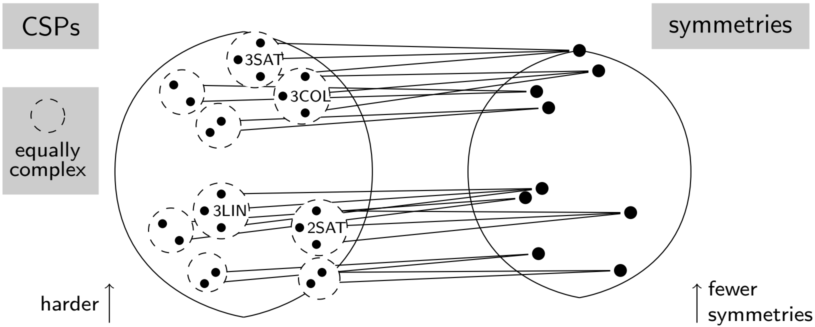


(1) polymorphisms

(2) systems of functional equations satisfied by polymorphisms

(3) minor conditions satisfied by polymorphisms

Where are the borderlines between complexity classes?

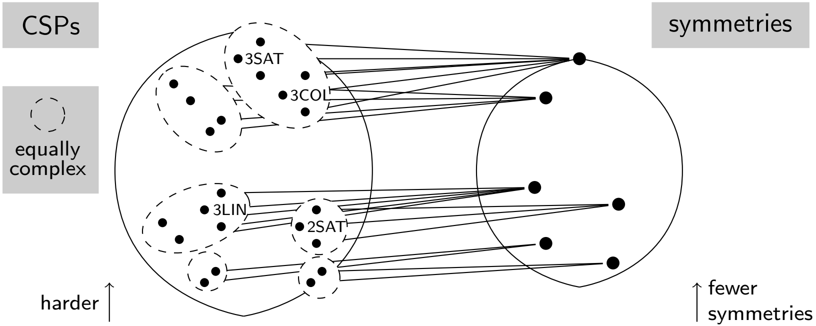


(1) polymorphisms

(2) systems of functional equations satisfied by polymorphisms

(3) minor conditions satisfied by polymorphisms

Where are the borderlines between complexity classes?

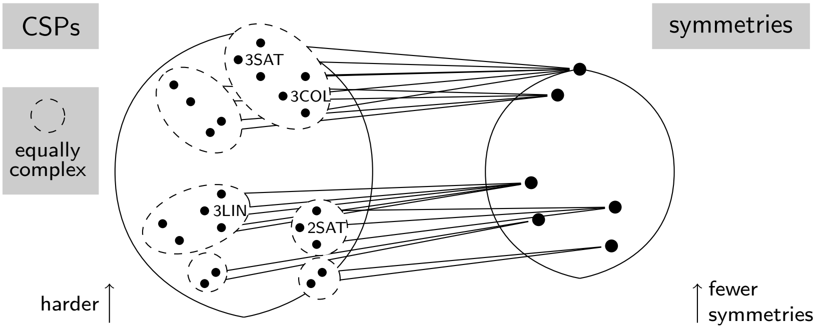


(1) polymorphisms

(2) systems of functional equations satisfied by polymorphisms

(3) minor conditions satisfied by polymorphisms

Where are the borderlines between complexity classes?

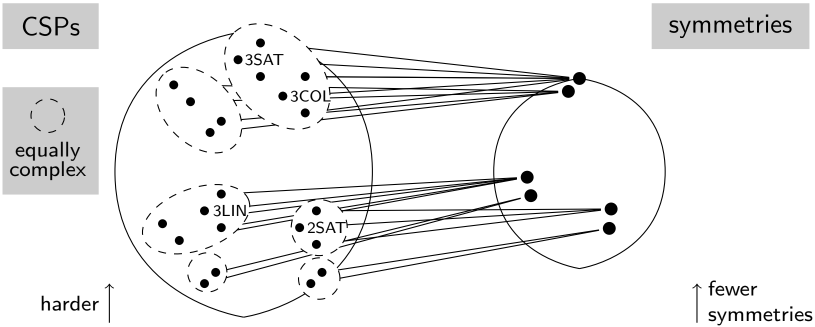


(1) polymorphisms

(2) systems of functional equations satisfied by polymorphisms

(3) minor conditions satisfied by polymorphisms

Where are the borderlines between complexity classes?

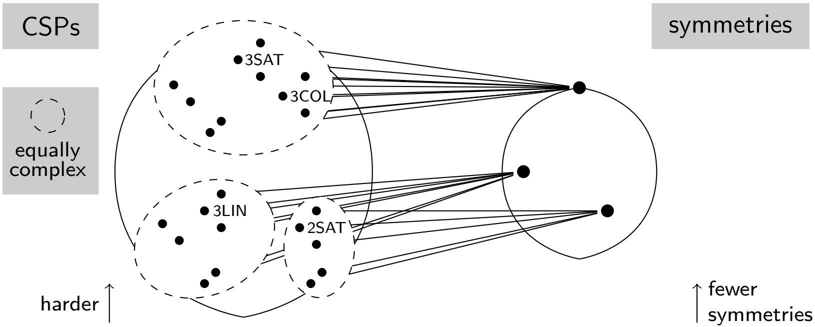


(1) polymorphisms

(2) systems of functional equations satisfied by polymorphisms

(3) minor conditions satisfied by polymorphisms

Where are the borderlines between complexity classes?



(1) polymorphisms

(2) systems of functional equations satisfied by polymorphisms

(3) minor conditions satisfied by polymorphisms

Where are the borderlines between complexity classes?

Minor condition is trivial:

satisfied in every $\mathrm{Pol}(\mathbb{A})$

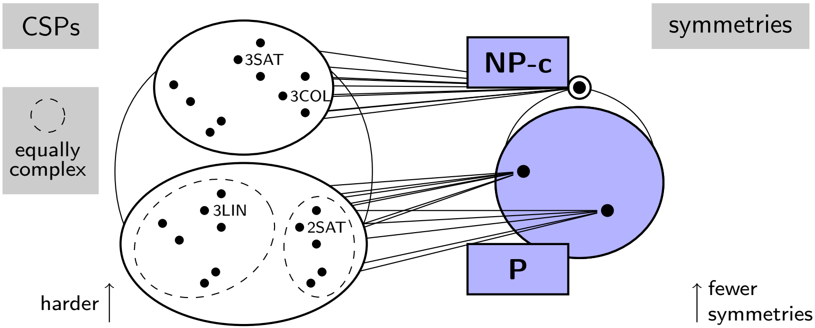
$=$ satisfied in $\mathcal{P}$, the set of projections on $\{0, 1\}$

Corollary

If $\mathrm{Pol}(\mathbb{A})$ *satisfies only trivial minor conditions,*
then $\mathrm{CSP}(\mathbb{A})$ *is NP-hard.*

Theorem ([Bulatov'17], [Zhuk'17])

If $\mathrm{Pol}(\mathbb{A})$ *satisfies some non-trivial minor condition,*
then $\mathrm{CSP}(\mathbb{A})$ *is in P.*



- only trivial minor conditions ⇒ NP-complete
- some nontrivial minor condition ⇒ P

Further steps?

(Barto,) Bulín, Krokhin, Opršal: Algebraic approach to promise constraint satisfaction

Definition ($\mathrm{MinorCond}(N, \mathcal{M})$)

Input: minor condition **X** with symbols of arity N
Answer Yes: **X** is trivial (=satisfied in $\mathcal{P}$)
Answer No: **X** not satisfied in $\mathcal{M}$

Theorem

Let $\mathcal{M} = \mathrm{Pol}(\mathbb{A})$*. The following computational problems are equivalent for a large enough* N*.*

(i) $\mathrm{CSP}(\mathbb{A})$
(ii) $\mathrm{MinorCond}(N, \mathcal{M})$

Consequence: 3rd step
Proof: direct, simple, known

Proof 1: Reduction from CSP

Given input of $\mathrm{CSP}(3\mathbb{NAE}_2)$, eg.

$$(\exists a, b, c, d)\ R(c, a, b) \wedge R(a, d, c)$$

transform it to a minor condition, eg.

$$\begin{aligned}
f_1(x_1, x_0, x_0, x_0, x_1, x_1) &= g_c(x_0, x_1)\\
f_1(x_0, x_1, x_0, x_1, x_0, x_1) &= g_a(x_0, x_1)\\
f_1(x_0, x_0, x_1, x_1, x_1, x_0) &= g_b(x_0, x_1)
\end{aligned}$$

$$\begin{aligned}
f_2(x_1, x_0, x_0, x_0, x_1, x_1) &= g_a(x_0, x_1)\\
f_2(x_0, x_1, x_0, x_1, x_0, x_1) &= g_d(x_0, x_1)\\
f_2(x_0, x_0, x_1, x_1, x_1, x_0) &= g_c(x_0, x_1)
\end{aligned}$$

"Yes input $\to$ Yes input": easy

"No input $\to$ No input": for contrapositive use $y \mapsto g_y(0, 1)$.

Given a minor condition, e.g.

$$f(x_1, x_2, x_1, x_3) = g(x_1, x_2, x_3)$$
$$h(x_3, x_1) = g(x_1, x_2, x_3)$$

- introduce variables f_{a_1,a_2,a_3,a_4} one for each $(a_1, \ldots, a_4) \in A^4$, h_{a_1,a_2}, and g_{a_1,a_2,a_3}.
- so evaluation of f's $\leftrightarrow$ function $f : A^4 \to A$
- express that f, g, h are polymorphisms (by constraints)
- merge variables to enforce the equations

The proof only uses bipartite minor conditions:

- Two disjoint set of symbols LHS, RHS.
- Each equation of the form

$$f(variables) = g(x_1, x_2, \ldots, x_N)$$

where $f \in LHS$ and $g \in RHS$

Remarks

How to show that $\mathcal{M} = \mathrm{Pol}(\mathbb{A})$ satisfies only trivial minor conditions?

Theorem

The following are equivalent

- *$\mathcal{M}$ satisfies only trivial minor conditions*
- *There is a mapping $\xi : \mathcal{M} \to \mathbb{N}$*
  - *if f is of arity n, then $\xi(f) \in \{1, 2, \dots, n\}$*
    *(**think**: an important coordinate of f)*
  - *ξ behaves nicely with minors, eg. if*

    $$f(x_3, x_2, x_1, x_2, x_2, x_1) = g(x_1, x_2, x_3)$$

    and $\xi(f) = 5$, then $\xi(g) = 2$.

How to devise algorithms if $\mathcal{M} = \mathrm{Pol}(\mathbb{A})$ satisfies some nontrivial minor condition?

Theorem

The following are equivalent.

- $\mathcal{M}$ *satisfies some nontrivial minor condition*
- $\mathcal{M}$ *satisfies, for some $n \geq 2$, the minor condition*

$$c(x_1, x_2, \ldots, x_n) = c(x_2, \ldots, x_n, x_1)$$

  [Barto, Kozik'12]

- *. . .*
- *. . . zillion other characterizations . . .*
- *. . .*

General problem: Given a structure $\mathfrak{A}$ and 1st order sentence ϕ (the same language), decide whether $\mathfrak{A}$ satisfies ϕ.

CSP

- fix a finite relational structure
- restrict to primitive positive (pp-) sentences

Another problem: Given a structure $\mathfrak{A}$ and 1st order sentence ϕ (different language), decide whether symbols in ϕ can be interpreted in $\mathfrak{A}$ so that $\mathfrak{A}$ satisfies ϕ.

Our case: solving functional equations over an algebra

- fix a finite algebraic structure
- restrict to universally quantified conjunction of (special) equations
- take a promise version