Checking quasi-identities and solving equations

Erhard Aichinger

Institute for Algebra
Johannes Kepler University Linz
Linz, Austria

AAA104, Blagoevgrad, February 2024

Supported by the Austrian Science Fund (FWF) : P33878

Quasi-identities

Example

Is every solution of

$$x^2 + y^2 = 2$$

also a solution of

Example

Is every solution of

$$x^2 + y^2 = 2$$

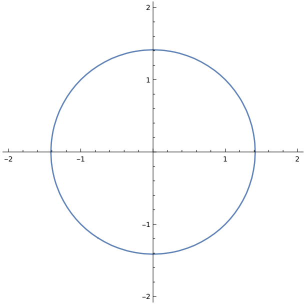
also a solution of

$$2x^2y - x^3 - x^2 - xy^2 + 2x + 2y^3 - y^2 - 4y + 2 = 0\,?$$

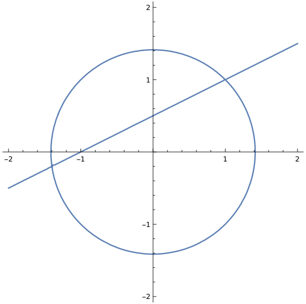
Example

Is every solution of $x^2 + y^2 = 2$ also a solution of
$2x^2y - x^3 - x^2 - xy^2 + 2x + 2y^3 - y^2 - 4y + 2 = 0$?

Hint 1:



$x^2 + y^2 = 2.$



$2x^2y - x^3 - x^2 - xy^2 + 2x + 2y^3 - y^2 - 4y + 2 = 0$

Example

Is every solution of $x^2 + y^2 = 2$ also a solution of $2x^2y - x^3 - x^2 - xy^2 + 2x + 2y^3 - y^2 - 4y + 2 = 0$?

Hint 2:

$$2x^2y - x^3 - x^2 - xy^2 + 2x + 2y^3 - y^2 - 4y + 2 = (-x + 2y - 1)(x^2 + y^2 - 2)$$

Example

Is every solution of $x^2 + y^2 = 2$ also a solution of
$2x^2y - x^3 - x^2 - xy^2 + 2x + 2y^3 - y^2 - 4y + 2 = 0$?

Hint 3: Try to find a counterexample with Mathematica.

$$\boldsymbol{P = -2 + x^2 + y^2;}$$

$$\boldsymbol{Q = 2 + 2x - x^2 - x^3 - 4y + 2x^2y - y^2 - xy^2 + 2y^3;}$$

GroebnerBasis$\boldsymbol{[\{P, Q * z - 1\}, \{x, y, z\}]}$

$\{1\}$

Quasi-identities in classical algebra

Theorem (Hilbert 1893).

For $f_1, \ldots, f_s, g \in \mathbb{C}[x_1, \ldots, x_n]$, the quasi-identity

$$\forall \boldsymbol{x} \in \mathbb{C}^n \ : \ f_1(\boldsymbol{x}) = \cdots = f_s(\boldsymbol{x}) = 0 \Longrightarrow g(\boldsymbol{x}) = 0$$

holds iff there are $a_1, \ldots, a_s \in \mathbb{C}[\boldsymbol{x}]$ and $r \in \mathbb{N}$ such that $g^r = a_1 f_1 + \cdots a_s f_s$.

Quasi-identities in classical algebra

Theorem (Hilbert 1893).

For $f_1, \dots, f_s, g \in \mathbb{C}[x_1, \dots, x_n]$, the quasi-identity

$$\forall \boldsymbol{x} \in \mathbb{C}^n \ : \ f_1(\boldsymbol{x}) = \cdots = f_s(\boldsymbol{x}) = 0 \Longrightarrow g(\boldsymbol{x}) = 0$$

holds iff there are $a_1, \dots, a_s \in \mathbb{C}[\boldsymbol{x}]$ and $r \in \mathbb{N}$ such that $g^r = a_1 f_1 + \cdots a_s f_s$.

Theorem (Terjanian 1966).

For $f_1, \dots, f_r, g \in \mathbb{F}_q[x_1, \dots, x_n]$, the quasi-identity

$$\forall \boldsymbol{x} \in \mathbb{F}_q{}^n \ : \ f_1(\boldsymbol{x}) = \cdots = f_s(\boldsymbol{x}) = 0 \Longrightarrow g(\boldsymbol{x}) = 0$$

holds in $\mathbb{F}_q$ iff there are $a_1, \dots, a_r, b_1, \dots, b_n \in \mathbb{F}_q[\boldsymbol{x}]$ such that

$$g = a_1 f_1 + \cdots a_r f_r + b_1 \cdot (x_1^q - x_1) + \cdots + b_n \cdot (x_n^q - x_n).$$

Quasi-identities in universal algebra

Quasi-identities in universal algebra

- $\mathbf{A}$ algebra, s_i, t_i, u, v terms.
- We ask whether $S = \{\boldsymbol{x} \in A^n \mid \bigwedge_{i \in \underline{k}} s_i(\boldsymbol{x}) = t_i(\boldsymbol{x})\}$ is contained in $U = \{\boldsymbol{x} \in A^n \mid u(\boldsymbol{x}) = v(\boldsymbol{x})\}$.
- This holds if the formula

$$\forall \boldsymbol{x} : \left(\bigwedge_{i \in \underline{k}} s_i(\boldsymbol{x}) = t_i(\boldsymbol{x})\right) \Longrightarrow u(\boldsymbol{x}) = v(\boldsymbol{x})$$

  holds in $\mathbf{A}$.
- Such a formula is called a *conditional identity* or *quasi-identity*.
- We want to determine the validity of this formula.

Quasi-identities in universal algebra

- $\mathbf{A}$ algebra, s_i, t_i, u, v terms.
- We ask whether $S = \{\boldsymbol{x} \in A^n \mid \bigwedge_{i \in \underline{k}} s_i^{\mathbf{A}}(\boldsymbol{x}) = t_i^{\mathbf{A}}(\boldsymbol{x})\}$ is contained in $U = \{\boldsymbol{x} \in A^n \mid u^{\mathbf{A}}(\boldsymbol{x}) = v^{\mathbf{A}}(\boldsymbol{x})\}$.
- This holds if the formula

$$\forall \boldsymbol{x} : \Big(\bigwedge_{i \in \underline{k}} s_i(\boldsymbol{x}) = t_i(\boldsymbol{x})\Big) \Longrightarrow u(\boldsymbol{x}) = v(\boldsymbol{x})$$

  holds in $\mathbf{A}$.
- Such a formula is called a *conditional identity* or *quasi-identity*.
- We want to determine the validity of this formula.

Quasi-identity validity

Let $\mathbf{A}$ be an algebra. QuasiIdVal($\mathbf{A}$) is the problem:

Given: A quasi-identity $\Phi := \forall \boldsymbol{x} \; : \; \left(\bigwedge_{i\in\underline{k}} s_i(\boldsymbol{x}) = t_i(\boldsymbol{x})\right) \implies u(\boldsymbol{x}) = v(\boldsymbol{x})$.
Here, s_i, t_i, u, v are terms in the language of $\mathbf{A}$ over the variables $\boldsymbol{x}$.

Asked: Does Φ hold in $\mathbf{A}$?

Quasi-identity validity

Let $\mathbf{A}$ be an algebra. QuasiIdVal($\mathbf{A}$) is the problem:

Given: A quasi-identity $\Phi := \forall \boldsymbol{x} \, : \, \left(\bigwedge_{i \in \underline{k}} s_i(\boldsymbol{x}) = t_i(\boldsymbol{x}) \right) \implies u(\boldsymbol{x}) = v(\boldsymbol{x})$.
Here, s_i, t_i, u, v are terms in the language of $\mathbf{A}$ over the variables $\boldsymbol{x}$.

Asked: Does Φ hold in $\mathbf{A}$?

Computational Complexity: For finite $\mathbf{A}$ of finite type, QuasiIdVal($\mathbf{A}$) is in co-**NP**:
$\boldsymbol{a} \in A^n$ witnesses failure of Φ if $\left(\bigwedge_{i \in \underline{k}} s_i^{\mathbf{A}}(\boldsymbol{a}) = t_i^{\mathbf{A}}(\boldsymbol{a}) \right) \wedge u^{\mathbf{A}}(\boldsymbol{a}) \neq v^{\mathbf{A}}(\boldsymbol{a})$.

Quasi-identity validity

Let $\mathbf{A}$ be an algebra. QuasiIdVal($\mathbf{A}$) is the problem:

Given: A quasi-identity $\Phi := \forall \boldsymbol{x} : \left(\bigwedge_{i \in \underline{k}} s_i(\boldsymbol{x}) = t_i(\boldsymbol{x})\right) \Longrightarrow u(\boldsymbol{x}) = v(\boldsymbol{x})$.
Here, s_i, t_i, u, v are terms in the language of $\mathbf{A}$ over the variables $\boldsymbol{x}$.

Asked: Does Φ hold in $\mathbf{A}$?

Computational Complexity: For finite $\mathbf{A}$ of finite type, QuasiIdVal($\mathbf{A}$) is in co-**NP**:
$\boldsymbol{a} \in A^n$ witnesses failure of Φ if $\left(\bigwedge_{i \in \underline{k}} s_i^{\mathbf{A}}(\boldsymbol{a}) = t_i^{\mathbf{A}}(\boldsymbol{a})\right) \wedge u^{\mathbf{A}}(\boldsymbol{a}) \neq v^{\mathbf{A}}(\boldsymbol{a})$.

Exponential time method: A quasi-identity of length ℓ contains at most ℓ different variables that can take at most $|A|^{\ell}$ values.

Question: For which algebras do we have faster methods (e.g. polynomial time)?

The complexity of quasi-identity validity

Quasi-identity validity and polynomial systems

Relations to other problems:

- If we can decide solvability of polynomial systems, then we can check the validity of quasi-identities.

Quasi-identity validity and polynomial systems

Relations to other problems:

- If we can decide solvability of polynomial systems, then we can check the validity of quasi-identities.
- We search for a counter-example: $\forall \boldsymbol{x} \; : \; \left(\bigwedge_{i \in \underline{k}} s_i(\boldsymbol{x}) = t_i(\boldsymbol{x}) \right) \implies u(\boldsymbol{x}) = v(\boldsymbol{x})$ holds iff for all $a, b \in A$ with $a \neq b$,

$$\bigwedge_{i \in \underline{k}} s_i(\boldsymbol{x}) = t_i(\boldsymbol{x}), \; u(\boldsymbol{x}) = a, \; v(\boldsymbol{x}) = b$$

  has no solution.
- These systems use constants: a and b.
  Therefore they are **polynomial systems** and not just **term systems**.

Quasi-identity validity and polynomial systems

Relations to other problems:

- If we can decide solvability of polynomial systems, then we can check the validity of quasi-identities.
- We search for a counter-example: $\forall \boldsymbol{x} : \left(\bigwedge_{i \in \underline{k}} s_i(\boldsymbol{x}) = t_i(\boldsymbol{x}) \right) \Longrightarrow u(\boldsymbol{x}) = v(\boldsymbol{x})$ holds iff for all $a, b \in A$ with $a \neq b$,

$$\bigwedge_{i \in \underline{k}} s_i(\boldsymbol{x}) = t_i(\boldsymbol{x}), \; u(\boldsymbol{x}) = a, \; v(\boldsymbol{x}) = b$$

  has no solution.
- These systems use constants: a and b.
  Therefore they are **polynomial systems** and not just **term systems**.
- **Conclusion:** $\text{QuasiIdVal}(\mathbf{A}) \leq_{\text{truth table}} \text{PolSysSat}(\mathbf{A})$.

Quasi-identity validity and systems of term equations

- If we can check the validity of quasi-identities, then we can decide solvability of term equations.

Quasi-identity validity and systems of term equations

- If we can check the validity of quasi-identities, then we can decide solvability of term equations.
- The system $s_1 = t_1, \ldots, s_k = t_k$ has no solution iff

$$s_1 = t_1 \wedge \ldots \wedge s_k = t_k \Longrightarrow y = z$$

is valid in $\mathbf{A}$. ($y, z \ldots$ new variables, $|A| > 1$).

Quasi-identity validity and systems of term equations

- If we can check the validity of quasi-identities, then we can decide solvability of term equations.
- The system $s_1 = t_1, \ldots, s_k = t_k$ has no solution iff

$$s_1 = t_1 \wedge \ldots \wedge s_k = t_k \Longrightarrow y = z$$

  is valid in $\mathbf{A}$. (y, z ...new variables, $|A| > 1$).
- **Conclusion:** co-$\textsc{TermSysSat}(\mathbf{A}) \leq_P \textsc{QuasiIdVal}(\mathbf{A})$.

Quasi-identity validity and checking term equivalence

- If we can check the validity of quasi-identities, we can check whether two terms induce the same function.

Quasi-identity validity and checking term equivalence

- If we can check the validity of quasi-identities, we can check whether two terms induce the same function.
- $\forall \boldsymbol{x} : s(\boldsymbol{x}) = t(\boldsymbol{x})$ is valid iff

$$y = y \Longrightarrow s(\boldsymbol{x}) = t(\boldsymbol{x})$$

  is valid in $\mathbf{A}$.

Quasi-identity validity and checking term equivalence

- If we can check the validity of quasi-identities, we can check whether two terms induce the same function.
- $\forall \boldsymbol{x} : s(\boldsymbol{x}) = t(\boldsymbol{x})$ is valid iff

$$y = y \Longrightarrow s(\boldsymbol{x}) = t(\boldsymbol{x})$$

  is valid in $\mathbf{A}$.
- **Conclusion:** $\textsc{TermEqv}(\mathbf{A}) \leq_P \textsc{QuasiIdVal}(\mathbf{A})$.

Quasi-identity validity: connections with well-studied problems.

Connections:

- $\textsc{QuasiIdVal}(\mathbf{A}) \leq_{\text{truth table}} \textsc{PolSysSat}(\mathbf{A})$.
- $\text{co-}\textsc{TermSysSat}(\mathbf{A}) \leq_P \textsc{QuasiIdVal}(\mathbf{A})$.
- $\textsc{TermEqv}(\mathbf{A}) \leq_P \textsc{QuasiIdVal}(\mathbf{A})$.

Quasi-identity validity: connections with well-studied problems.

Connections:

- $\textsc{QuasiIdVal}(\mathbf{A}) \leq_{\text{truth table}} \textsc{PolSysSat}(\mathbf{A})$.
- co-$\textsc{TermSysSat}(\mathbf{A}) \leq_P \textsc{QuasiIdVal}(\mathbf{A})$.
- $\textsc{TermEqv}(\mathbf{A}) \leq_P \textsc{QuasiIdVal}(\mathbf{A})$.
- In 2004, M. Volkov constructed a 10-element semigroup $\mathbf{Q}$ with $\textsc{TermEqv}(\mathbf{Q}) \in \mathbf{P}$, and $\textsc{QuasiIdVal}(\mathbf{Q})$ co-**NP**-complete because it solves 3-$\textsc{Colorability}$ for graphs.

Quasi-identity validity: connections with well-studied problems.

Let $\mathbf{A}$ be an algebra with a Mal'cev term.

Consequences:

- $\mathbf{A}$ is abelian $\implies$ QuasiIdVal($\mathbf{A}$) $\in \mathbf{P}$.
  (Reason: PolSysSat, which is analyzed in [Larose, Zádori 2006])

Quasi-identity validity: connections with well-studied problems.

Let $\mathbf{A}$ be an algebra with a Mal'cev term.

Consequences:

- $\mathbf{A}$ is abelian $\implies$ $\textsc{QuasiIdVal}(\mathbf{A}) \in \mathbf{P}$.
  (Reason: $\textsc{PolSysSat}$, which is analyzed in [Larose, Zádori 2006])
- $\mathrm{Core}(\mathbf{A})$ is nonabelian $\implies$ $\textsc{QuasiIdVal}(\mathbf{A})$ is co-**NP**-complete.
  (Reason: $\textsc{TermSysSat}$, which is analyzed in [Mayr 2023])

Quasi-identity validity: connections with well-studied problems.

Let $\mathbf{A}$ be an algebra with a Mal'cev term.

Consequences:

- $\mathbf{A}$ is abelian $\implies$ $\textsc{QuasiIdVal}(\mathbf{A}) \in \mathbf{P}$.
  (Reason: $\textsc{PolSysSat}$, which is analyzed in [Larose, Zádori 2006])
- $\mathrm{Core}(\mathbf{A})$ is nonabelian $\implies$ $\textsc{QuasiIdVal}(\mathbf{A})$ is co-**NP**-complete.
  (Reason: $\textsc{TermSysSat}$, which is analyzed in [Mayr 2023])
- $\mathbf{A}$ non-solvable group $\implies$ $\textsc{QuasiIdVal}(\mathbf{A})$ is co-**NP**-complete.
  (Reason: $\textsc{TermEqv}$, which is analyzed in [Horváth, Lawrence, Mérai, Szabó 2007])

Quasi-identity validity: connections with well-studied problems.

Let **A** be an algebra with a Mal'cev term.

Consequences:

- **A** is abelian $\implies$ QuasiIdVal(**A**) $\in$ **P**.
  (Reason: PolSysSat, which is analyzed in [Larose, Zádori 2006])
- Core(**A**) is nonabelian $\implies$ QuasiIdVal(**A**) is co-**NP**-complete.
  (Reason: TermSysSat, which is analyzed in [Mayr 2023])
- **A** non-solvable group $\implies$ QuasiIdVal(**A**) is co-**NP**-complete.
  (Reason: TermEqv, which is analyzed in [Horváth, Lawrence, Mérai, Szabó 2007])

Open: nonabelian nilpotent groups, nonzero nilpotent rings.

A reduction of graph coloring to quasi-identities

Quasi-identity validity

Theorem Aichinger, Grünbacher, STACS 2023

$\mathbf{A}$ finite algebra of finite type with a Mal'cev term. Then

1. $\textsc{QuasiIdVal}(\mathbf{A}) \in \mathbf{P}$ if $\mathbf{A}$ is abelian.
2. $\textsc{QuasiIdVal}(\mathbf{A})$ is co-**NP**-complete if $\mathbf{A}$ is nonabelian.

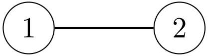
New content: item (2).

Proof idea: we reduce the H-coloring problem to $\textsc{QuasiIdVal}(\mathbf{A})$.

H-coloring of graphs

H-COLORING:
Given: a graph G.
Asked: Is there a graph homomorphism h from G to H ($G \to H$)?

- $H = K_2$:

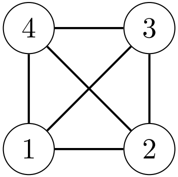


$G \to H$ iff G is bipartite: edges in G only go from $h^{-1}(\{1\})$ to $h^{-1}(\{2\})$.

H-coloring of graphs

H-COLORING:
Given: a graph G.
Asked: Is there a graph homomorphism h from G to H ($G \to H$)?

- $H = K_4$:

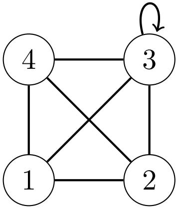


$G \to H$ if the vertices of G can be coloured with 4 colors such that no adjacent vertices have the same colour.

H-coloring of graphs

H-COLORING:
Given: a graph G.
Asked: Is there a graph homomorphism h from G to H ($G \to H$)?

- H a graph with loops:



$G \to H$ holds for every graph G: use $h(v) = 3$ for each vertex v of G.

Theorem Hell, Nešetřil 1990.

Let H be a finite loopless graph that contains a triangle. Then H-COLORING is **NP**-complete.

A consequence stated in CSP-language:

Theorem

Let $\mathbb{H} = (H, \rho)$ be a relational structure with an antireflexive and symmetric binary relation ρ.

If $\mathbb{H}$ has $\mathbb{K}_3 = (\{1, 2, 3\}; \neq)$ as a substructure, then $\mathrm{CSP}(\mathbb{H})$ is **NP**-complete.

Proof of the Theorem

Plan:

- We want to prove that checking the validity of quasi-identities of $\mathbf{R} := (3\mathbb{Z}_{27}, +, -, \cdot, 0)$ is co-**NP**-complete.
- We will show: there is a graph H such that

$$\text{for every graph } G: \quad G \to H \iff \text{the quasi-identity } \Phi(G) \text{ is not valid.}$$

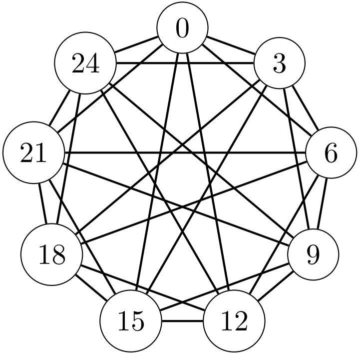
- This will imply that $\textsc{QuasiIdVal}(\mathbf{R})$ is co-**NP**-complete.

Details:

- $R = \{[0]_{27}, [3]_{27}, \dots, [24]_{27}\}$.
- H is the *"difference graph"* or *"apartness graph"* on R: (r, s) is an edge if $r - s \notin \{[0]_{27}, [9]_{27}, [18]_{27}\}$.

Proof of the Theorem

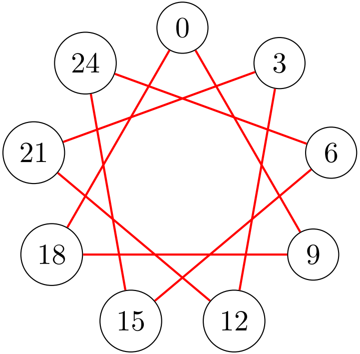
The graph H for $3\mathbb{Z}_{27}$



$$E(H) = \{(x, y) \mid x - y \notin \{0, 9, 18\}\}.$$

Proof of the Theorem

The graph H for $3\mathbb{Z}_{27}$

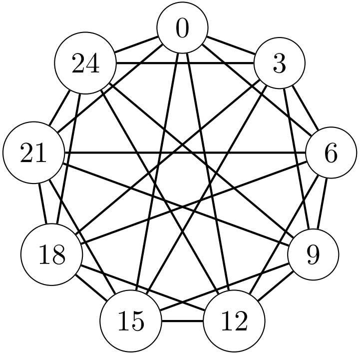


$$E(H) = \{(x, y) \mid x - y \notin \{0, 9, 18\}\}.$$

- non-edges of H

Proof of the Theorem

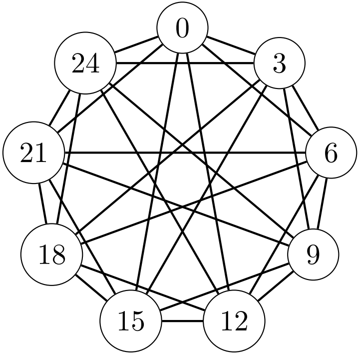
The graph H for $3\mathbb{Z}_{27}$



- G graph. We want to find out whether $G \to H$ **using a quasi-identity on R**.
- $\Phi = \left(\bigwedge_{(u,v)\in E(G)} a = z_{u,v} \cdot (x_u - x_v) \right) \Rightarrow a = 0.$

Proof of the Theorem

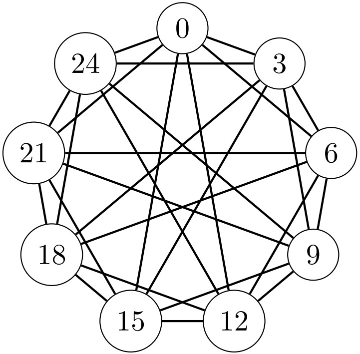
The graph H for $3\mathbb{Z}_{27}$



- G graph. We want to find out whether $G \to H$ **using a quasi-identity on R**.
- $\Phi = \left(\bigwedge_{(u,v)\in E(G)} a = z_{u,v} \cdot (x_u - x_v) \right) \Rightarrow a = 0.$
- Suppose Φ is invalid. Then $a \neq 0$.
- Let $(u, v) \in E(G)$. Then $x_u - x_v \notin \{0, 9, 18\}$.
- Thus (x_u, x_v) is an edge of H.
- $u \mapsto x_u$ is a homomorphism from G to H.
- Hence if Φ is invalid, $G \to H$.

Proof of the Theorem

The graph H for $3\mathbb{Z}_{27}$



- G graph. We want to find out whether $G \to H$ **using a quasi-identity on R**.
- $\Phi = \Big(\bigwedge_{(u,v)\in E(G)} a = z_{u,v} \cdot (x_u - x_v) \Big) \Rightarrow a = 0.$
- Suppose $G \to H$.
- …
- This is a counterexample to Φ.
- Hence Φ is invalid.

Proof of the Theorem

- Hence Φ is not valid iff $G \to H$.
- H-coloring is **NP**-complete [Hell, Nešetřil 1990].
- Thus QuasiIdVal($\mathbf{R}$) is co-**NP**-complete.

Proof for Mal'cev algebras

Theorem

Let $\mathbf{A}$ be a finite nonabelian algebra of finite type with a Mal'cev term. Then QuasiIdVal($\mathbf{A}$) is co-**NP**-complete.

- Instead of the ring multiplication, use commutators [Smith 1976, Hagemann, Herrmann 1979].
- This works for subdirectly irreducible $\mathbf{A}$.
- For arbitrary $\mathbf{A}$, use "difference graphs" for several congruences of $\mathbf{A}$.
- Order these graphs and pick a maximal one.
- Erhard Aichinger and Simon Grünbacher. *The Complexity of Checking Quasi-Identities over Finite Algebras with a Mal'cev Term*, STACS 2023.

Additional material on this topic that was not presented in the talk at AAA104:

Systems of term equations

Systems of term equations

Let $\mathbf{A}$ be an algebra.
$\textsc{TermSysSat}(\mathbf{A})$ is the following problem:

Given:
Terms $s_1(x_1, \ldots, x_n), t_1(x_1, \ldots, x_n), \ldots, s_k(x_1, \ldots, x_n), t_k(x_1, \ldots, x_n)$.

Asked:
Is there $\boldsymbol{a} \in A^n$ with $s_1^{\mathbf{A}}(\boldsymbol{a}) = t_1^{\mathbf{A}}(\boldsymbol{a}), \ldots, s_k^{\mathbf{A}}(\boldsymbol{a}) = t_k^{\mathbf{A}}(\boldsymbol{a})$?

Computational complexity of TermSysSat($\mathbf{A}$)

One can solve the equations by solving a *constraint satisfaction problem*.

Idea: (Larose, Zádori 2006)

Instead of solving

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

solve

$$(x_1, x_2, y_1) \in g^\circ, (y_1, y_2) \in f^\circ, (x_1, y_2) \in f^\circ, \text{ where}$$

$$g^\circ = \{(a_1, a_2, b) \in A^3 \mid g(a_1, a_2) = b\}$$

is the *graph* of g.

This reduces TermSysSat($A; f, g$) to CSP($A; f^\circ, g^\circ$).

Computational complexity of $\textsc{TermSysSat}(\mathbf{A})$

For an algebra $\mathbf{A} = (A; F)$, let $\mathbf{A}^\circ := (A; \{f^\circ \mid f \in F\})$.

As a consequence of the Bulatov-Zhuk-Dichotomy (2017) (in the form of Barto, Krokhin, Willard (2017)), one obtains:

Theorem (cf. [Mayr, MFCS 2023]).

(Assume $\mathbf{P} \neq \mathbf{NP}$).
Let $\mathbf{A}$ be a finite algebra. Then $\textsc{TermSysSat}(\mathbf{A}) \in \mathbf{P} \iff \mathbf{A}^\circ$ has a (not necessarily idempotent) Taylor polymorphism.

Otherwise $\textsc{TermSysSat}(\mathbf{A})$ is $\mathbf{NP}$-complete.

Computational complexity of TermSysSat($\mathbf{A}$)

Question: Algebraic description when $\mathbf{A}^\circ$ has a (not necessarily idempotent) Taylor polymorphism.

Definition. Let $\mathbf{A}$ be a finite algebra.
Core($\mathbf{A}$) is a minimal endomorphic image of $\mathbf{A}$ w.r.t $\subseteq$.
(Defined up to isomorphism)

Examples.

- $\mathbf{G}$ group. $\mathrm{Core}(\mathbf{G}) = \{1\}$.
- $\mathbf{G}$ group. $\mathbf{G}^* := (G; *, {}^{-1}, (c_g)_{g\in G})$ its expansions with all constants from G. Then $\mathrm{Core}(\mathbf{G}^*) = G$.
- $\mathrm{Core}((S_5; \circ, {}^{-1}, \underbrace{\mathrm{id}, (1\ 2)}_{\text{nullary}})) = \{\mathrm{id}, (1\ 2)\}$.

Computational complexity of $\textsc{TermSysSat}(\mathbf{A})$

Theorem Larose, Zádori 2006

Let $\mathbf{A}$ be a finite algebra in a congruence modular variety. TFAE:

1. $\textsc{PolSysSat}(\mathbf{A}) = \textsc{TermSysSat}(\mathbf{A}^*) \in \mathbf{P}$.
2. $\mathbf{A}$ is abelian.

Theorem Mayr 2023

Let $\mathbf{A}$ be a finite algebra in a congruence modular variety. TFAE:

1. $\textsc{TermSysSat}(\mathbf{A}) \in \mathbf{P}$.
2. $\mathrm{Core}(\mathbf{A})$ is abelian.

Both results also hold also if $1 \notin \mathrm{typ}(V(\mathbf{A}))$ and $5 \notin \mathrm{typ}(\{\mathbf{A}\})$.

TermSysSat($\mathbf{A}$) vs. PolSysSat($\mathbf{A}$)

Theorem Mayr 2023.

Let $\mathbf{A}$ be a finite algebra of finite type. The following three problems are reducible to each other in constant time:

1. TermSysSat($\mathbf{A}$).
2. TermSysSat(Core($\mathbf{A}$)).
3. PolSysSat(Core($\mathbf{A}$)).

The meta-problem for systems of term equations

The meta-problem for TermSysSat

Meta-problem for TermSysSat (Assume $\mathbf{P} \neq \mathbf{NP}$)

Given: $\mathbf{A} = (A; f_1, \ldots, f_k)$

Asked: Is $\textsc{TermSysSat}(\mathbf{A}) \in \mathbf{P}$?

The meta-problem for TermSysSat

Meta-problem for TermSysSat (Assume $\mathbf{P} \neq \mathbf{NP}$)

Given: $\mathbf{A} = (A; f_1, \ldots, f_k)$

Asked: Is TermSysSat($\mathbf{A}$) $\in \mathbf{P}$?

Asked: Does Core($\mathbf{A}^\circ$) have a Siggers polymorphism?

The meta-problem for TermSysSat

Meta-problem for TermSysSat (Assume $\mathbf{P} \neq \mathbf{NP}$)

Given: $\mathbf{A} = (A; f_1, \ldots, f_k)$

Asked: Is TermSysSat($\mathbf{A}$) $\in \mathbf{P}$?

Asked: Does Core($\mathbf{A}^\circ$) have a Siggers polymorphism?

In cm varieties: **Asked:** Does $\mathbf{A}$ have an abelian core?

Theorem Mayr 2023

There is a *quasi-polynomial* algorithm that decides whether a given finite $\mathbf{A}$ in a cm variety has an abelian core.

$q(n)$ is *quasi-polynomial* if $\exists c, d, N > 0\ \forall n \geq N\ :\ q(n) \leq c2^{\log(n)^d}$.

Solving systems of term equations over modules

Solving TermSysSat($\mathbf{A}$)

Let $\mathbf{A}$ be an $\mathbf{R}$-module.

- The polynomial algorithm provided by the theory uses the Bulatov-Dalmau-algorithm (2006) to solve instances of CSP($\mathbf{A}^\circ$), which has the Mal'cev term of $\mathbf{A}$ as a polymorphism.
- In practice, *Hermite-decomposition* is useful.

Solving $\textsc{TermSysSat}(\mathbf{A})$

We solve

$$\begin{pmatrix} 10 & 16 & 0 \\ 15 & 24 & 30 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 4 \\ 66 \end{pmatrix}$$

over $\mathbb{Z}$.

Solving TermSysSat($\mathbf{A}$)

We solve

$$\begin{pmatrix} 10 & 16 & 0 \\ 15 & 24 & 30 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 4 \\ 66 \end{pmatrix}$$

over $\mathbb{Z}$. To this end, we compute a $\mathbb{Z}$-Basis of the row module of

$$\begin{pmatrix} 4 & 66 & 1 & 0 & 0 & 0 \\ 10 & 15 & 0 & 1 & 0 & 0 \\ 16 & 24 & 0 & 0 & 1 & 0 \\ 0 & 30 & 0 & 0 & 0 & 1 \end{pmatrix}$$

using the Hermite normal form (1851, polynomial time since 1979).

Solving TermSysSat(**A**)

We solve

$$\begin{pmatrix} 10 & 16 & 0 \\ 15 & 24 & 30 \end{pmatrix} \cdot \left(\begin{smallmatrix} x \\ y \\ z \end{smallmatrix}\right) = \begin{pmatrix} 4 \\ 66 \end{pmatrix}$$

over $\mathbb{Z}$. We have

$$\operatorname{row}(\begin{pmatrix} 4 & 66 & 1 & 0 & 0 & 0 \\ 10 & 15 & 0 & 1 & 0 & 0 \\ 16 & 24 & 0 & 0 & 1 & 0 \\ 0 & 30 & 0 & 0 & 0 & 1 \end{pmatrix}) = \operatorname{row}(\begin{pmatrix} 2 & 3 & 0 & 5 & -3 & 0 \\ 0 & 30 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 6 & -4 & -2 \\ 0 & 0 & 0 & 8 & -5 & 0 \end{pmatrix})$$

and thus $S = \{(-6, 4, 2) + t\,(8, -5, 0) \mid t \in \mathbb{Z}\}$.

Solving CSP's through equations

Theorem.
For every finite relational structure $\mathbb{D}$ of finite type, there is a finite algebra $\mathbf{A}(\mathbb{D})$ such that CSP($\mathbb{D}$) and $\textsc{TermSysSat}(\mathbf{A}(\mathbb{D}))$ are polynomial time reducible to each other.

1. Klíma, Tesson, Thérien 2007:
   Assume $\mathbb{D} = (D, \rho)$ is a digraph. $\mathbf{A}(\mathbb{D})$ is a semigroup with $5|D| + |\rho| + 1$ elements that satisfies $x^2 \approx x$ and $xyz \approx yxz$.
2. Broniek 2015:
   Assume $\mathbb{D} = (D, R)$ with $R \subseteq D^r$. $\mathbf{A}(\mathbb{D})$ is a *unary* algebra with $|D| + |R| + 2$ elements and $r + 4$ unary operations.