The Sorrows of a Young Algebraist

Tomáš Nagy

Jagiellonian University

joint work with
J. B. Brunar, J. W. Goethe, M. Kozik, and M. Pinsker

AAA 108, Wien, 8th February 2026

Neither the author nor the co-authors are responsible
for any consequences or side effects of this talk.

Cast

Charlotte: CSP
(Johanna Beate Brunar)

Cast

Charlotte: CSP
(Johanna Beate Brunar)

Werther (aka the Young Algebraist): Universal Algebra
(Philipp Alexander Grzywaczyk)

Cast

Charlotte: CSP
(Johanna Beate Brunar)

Werther (aka the Young Algebraist): Universal Algebra
(Philipp Alexander Grzywaczyk)

Albert (Charlotte's fiancé and later husband): Computer Science
(any volunteers?)

Finite-domain CSP $\leftrightarrow$ universal algebra

Finite-domain CSP $\leftrightarrow$ universal algebra

Congruences of algebras $\leftrightarrow$ identities $\leftrightarrow$ algorithms (e.g., Bulatov, Dalmau)

Finite-domain CSP $\leftrightarrow$ universal algebra

Congruences of algebras $\leftrightarrow$ identities $\leftrightarrow$ algorithms (e.g., Bulatov, Dalmau)

New identities proven algebraically (absorption, centers, . . .)

Finite-domain CSP $\leftrightarrow$ universal algebra

Congruences of algebras $\leftrightarrow$ identities $\leftrightarrow$ algorithms (e.g., Bulatov, Dalmau)

New identities proven algebraically (absorption, centers, ...)

$\rightsquigarrow$ the Bulatov-Zhuk dichotomy theorem relies on algebraic machinery

Does CSP Still Need Algebra?

After the dichotomy proof: Focus shifted to variants of CSP (PCSPs, Quantum CSPs, VCSPs, ...)

Does CSP Still Need Algebra?

After the dichotomy proof: Focus shifted to variants of CSP (PCSPs, Quantum CSPs, VCSPs, . . .)

Algebraic methods of limited utility,

Does CSP Still Need Algebra?

After the dichotomy proof: Focus shifted to variants of CSP (PCSPs, Quantum CSPs, VCSPs, . . .)

Algebraic methods of limited utility, instead

- Hardcore CS,

Does CSP Still Need Algebra?

After the dichotomy proof: Focus shifted to variants of CSP (PCSPs, Quantum CSPs, VCSPs, . . .)

Algebraic methods of limited utility, instead

- Hardcore CS,
- Topology,

Does CSP Still Need Algebra?

After the dichotomy proof: Focus shifted to variants of CSP (PCSPs, Quantum CSPs, VCSPs, . . .)

Algebraic methods of limited utility, instead

- Hardcore CS,
- Topology,
- Category theory,

Does CSP Still Need Algebra?

After the dichotomy proof: Focus shifted to variants of CSP (PCSPs, Quantum CSPs, VCSPs, . . .)

Algebraic methods of limited utility, instead

- Hardcore CS,
- Topology,
- Category theory,
- . . .

Infinite-domain CSP
Algebraic approach generalises
(complexity depends only on polymorphisms)

Infinite-domain CSP
Algebraic approach generalises
(complexity depends only on polymorphisms)

Some identities lift (6-ary pseudo-Siggers)

Infinite-domain CSP
Algebraic approach generalises
(complexity depends only on polymorphisms)

Some identities lift (6-ary pseudo-Siggers)

New algebraic theory developed recently
(smooth approximations; Mottet, Pinsker, 2022)

The Feet of All the Horses

- The most useful identities do not lift (cyclic, WNUs unsure)

The Feet of All the Horses

- The most useful identities do not lift (cyclic, WNUs unsure)
- WLOG assumptions do not generalise (idempotency)

The Feet of All the Horses

- The most useful identities do not lift (cyclic, WNUs unsure)
- WLOG assumptions do not generalise (idempotency)
- Unclear how to lift most of the notions (absorption, . . .)

- The most useful identities do not lift (cyclic, WNUs unsure)
- WLOG assumptions do not generalise (idempotency)
- Unclear how to lift most of the notions (absorption, . . .)
- Important properties not characterised by identities (local consistency)

The Feet of All the Horses

- The most useful identities do not lift (cyclic, WNUs unsure)
- WLOG assumptions do not generalise (idempotency)
- Unclear how to lift most of the notions (absorption, . . .)
- Important properties not characterised by identities (local consistency)

$\Rightarrow$ Most of the finite-domain methods cannot be lifted

The programme of the jealous husband:

- Remove algebra from the finite proofs

The programme of the jealous husband:

- Remove algebra from the finite proofs
- Lift them to the infinite!

The programme of the jealous husband:

- Remove algebra from the finite proofs
- Lift them to the infinite!

Used successfully in several cases:

- Pseudo-loop lemmata (Bodor, Kozik, Mottet, Pinsker, 2023; Brunar, Kozik, Pinsker, N., 2025)

The programme of the jealous husband:

- Remove algebra from the finite proofs
- Lift them to the infinite!

Used successfully in several cases:

- Pseudo-loop lemmata (Bodor, Kozik, Mottet, Pinsker, 2023; Brunar, Kozik, Pinsker, N., 2025)
- FO vs L-hard dichotomy (Dorochko, Wrona, 2026)

The programme of the jealous husband:

- Remove algebra from the finite proofs
- Lift them to the infinite!

Used successfully in several cases:

- Pseudo-loop lemmata (Bodor, Kozik, Mottet, Pinsker, 2023; Brunar, Kozik, Pinsker, N., 2025)
- FO vs L-hard dichotomy (Dorochko, Wrona, 2026)

More to come? Pseudo-identities, local consistency, . . .

6-ary Siggers identity

$$s(x, y, z, x, y, z) \approx s(y, z, x, z, x, y)$$

6-ary Siggers identity

$$s(x, y, z, x, y, z) \approx s(y, z, x, z, x, y)$$

Hell-Nešetřil.

Every non-bipartite graph which does not pp-construct $\mathbb{K}_3$ contains a loop.

6-ary Siggers identity

$$s(x,y,z,x,y,z) \approx s(y,z,x,z,x,y)$$

Hell-Nešetřil.

Every non-bipartite graph which does not pp-construct $\mathbb{K}_3$ contains a loop.

4-ary Siggers identity

$$s(a,r,e,a) \approx s(r,a,r,e)$$

6-ary Siggers identity

$$s(x,y,z,x,y,z) \approx s(y,z,x,z,x,y)$$

Hell-Nešetřil.

Every non-bipartite graph which does not pp-construct $\mathbb{K}_3$ contains a loop.

4-ary Siggers identity

$$s(a,r,e,a) \approx s(r,a,r,e)$$

Barto-Kozik-Niven.

Every smooth digraph of al 1 which does not pp-construct $\mathbb{K}_3$ contains a loop.

$\mathbb{G}$ – smooth ω-categorical digraph

6-ary pseudo-Siggers identity (Barto, Pinsker, 2016)

$$\begin{aligned} & u \circ s(x, y, z, x, y, z) \\ \approx \; & v \circ s(y, z, x, z, x, y) \end{aligned}$$

$\mathbb{G}$ – smooth ω-categorical digraph

6-ary pseudo-Siggers identity (Barto, Pinsker, 2016)

$$u \circ s(x, y, z, x, y, z) \approx v \circ s(y, z, x, z, x, y)$$

Bodor, Kozik, Mottet, Pinsker.

If $\mathbb{G}/\operatorname{Aut}(\mathbb{G})$ non-bipartite graph which does not pp-construct $\mathbb{K}_3$, then $\mathbb{G}/\operatorname{Aut}(\mathbb{G})$ contains a loop.

$\mathbb{G}$ – smooth ω-categorical digraph

6-ary pseudo-Siggers identity (Barto, Pinsker, 2016)

$$u \circ s(x, y, z, x, y, z) \approx v \circ s(y, z, x, z, x, y)$$

4-ary pseudo-Siggers identity (???)

$$u \circ s(a, r, e, a) \approx v \circ s(r, a, r, e)$$

Bodor, Kozik, Mottet, Pinsker.

If $\mathbb{G}/\operatorname{Aut}(\mathbb{G})$ non-bipartite graph which does not pp-construct $\mathbb{K}_3$, then $\mathbb{G}/\operatorname{Aut}(\mathbb{G})$ contains a loop.

$\mathbb{G}$ – smooth ω-categorical digraph

6-ary pseudo-Siggers identity (Barto, Pinsker, 2016)

$$u \circ s(x, y, z, x, y, z) \approx v \circ s(y, z, x, z, x, y)$$

Bodor, Kozik, Mottet, Pinsker.

If $\mathbb{G}/\operatorname{Aut}(\mathbb{G})$ non-bipartite graph which does not pp-construct $\mathbb{K}_3$, then $\mathbb{G}/\operatorname{Aut}(\mathbb{G})$ contains a loop.

4-ary pseudo-Siggers identity (???)

$$u \circ s(a, r, e, a) \approx v \circ s(r, a, r, e)$$

?

The Dream.

$\mathbb{G}$ smooth ω-categorical digraph, $\mathbb{G}/\operatorname{Aut}(\mathbb{G})$ has algebraic length 1, and $\mathbb{G}$ does not pp-construct $\mathbb{K}_3 \Rightarrow \mathbb{G}/\operatorname{Aut}(\mathbb{G})$ contains a loop.

The Dream.

$\mathbb{G}$ smooth ω-categorical digraph, $\mathbb{G}/\operatorname{Aut}(\mathbb{G})$ has algebraic length 1, and $\mathbb{G}$ does not pp-construct $\mathbb{K}_3 \Rightarrow \mathbb{G}/\operatorname{Aut}(\mathbb{G})$ contains a loop.

Motivation:

- **Algebra:** 4-ary pseudo-Siggers and further identities,

The Dream.

$\mathbb{G}$ smooth ω-categorical digraph, $\mathbb{G}/\operatorname{Aut}(\mathbb{G})$ has algebraic length 1, and $\mathbb{G}$ does not pp-construct $\mathbb{K}_3 \Rightarrow \mathbb{G}/\operatorname{Aut}(\mathbb{G})$ contains a loop.

Motivation:

- **Algebra:** 4-ary pseudo-Siggers and further identities,
- **CS:** Hardness criteria for digraphs:
  $\mathbb{G}/\operatorname{Aut}(\mathbb{G})$ does not contain a loop $\Rightarrow \operatorname{CSP}(\mathbb{G})$ NP-hard

The Dream.

$\mathbb{G}$ smooth ω-categorical digraph, $\mathbb{G}/\operatorname{Aut}(\mathbb{G})$ has algebraic length 1, and $\mathbb{G}$ does not pp-construct $\mathbb{K}_3 \Rightarrow \mathbb{G}/\operatorname{Aut}(\mathbb{G})$ contains a loop.

Motivation:

- **Algebra:** 4-ary pseudo-Siggers and further identities,
- **CS:** Hardness criteria for digraphs:
  $\mathbb{G}/\operatorname{Aut}(\mathbb{G})$ does not contain a loop $\Rightarrow \operatorname{CSP}(\mathbb{G})$ NP-hard
- Development of new techniques

The Dream.

$\mathbb{G}$ smooth ω-categorical digraph, $\mathbb{G}/\operatorname{Aut}(\mathbb{G})$ has algebraic length 1, and $\mathbb{G}$ does not pp-construct $\mathbb{K}_3 \Rightarrow \mathbb{G}/\operatorname{Aut}(\mathbb{G})$ contains a loop.

Motivation:

- **Algebra:** 4-ary pseudo-Siggers and further identities,
- **CS:** Hardness criteria for digraphs:
  $\mathbb{G}/\operatorname{Aut}(\mathbb{G})$ does not contain a loop $\Rightarrow \operatorname{CSP}(\mathbb{G})$ NP-hard
- Development of new techniques

For $\mathbb{G}$ finite proven by Bodor, Kozik, Mottet, Pinsker, 2023

CS view on CSP:
$\mathrm{CSP}(\mathbb{G})$: Find a homomorphism h from an input graph $\mathbb{I}$ to $\mathbb{G}$
“colour the vertices of $\mathbb{I}$ by the vertices of $\mathbb{G}$”

CS view on CSP:
$\mathrm{CSP}(\mathbb{G})$: Find a homomorphism h from an input graph $\mathbb{I}$ to $\mathbb{G}$
"colour the vertices of $\mathbb{I}$ by the vertices of $\mathbb{G}$"

Natural requirement:
List homomorphism: For every vertex v of $\mathbb{I}$, prescribe a list $L(v)$ of admissible vertices of $\mathbb{G}$, i.e., require $h(v) \in L(v)$

$\rightsquigarrow$ $\mathrm{CSP}(\mathbb{G}+$ all subsets of vertices$)$.

CS view on CSP:
$\mathrm{CSP}(\mathbb{G})$: Find a homomorphism h from an input graph $\mathbb{I}$ to $\mathbb{G}$
"colour the vertices of $\mathbb{I}$ by the vertices of $\mathbb{G}$"

Natural requirement:
List homomorphism: For every vertex v of $\mathbb{I}$, prescribe a list $L(v)$ of admissible vertices of $\mathbb{G}$, i.e., require $h(v) \in L(v)$

$\rightsquigarrow \mathrm{CSP}(\mathbb{G}+$ all subsets of vertices$)$.

In the infinite: Vertices of $\mathbb{G} \rightsquigarrow$ orbits of elements under $\mathrm{Aut}(\mathbb{G})$

Theorem [Brunar, Kozik, N., Pinsker]

$\mathbb{G}$ smooth ω-categorical digraph, $\mathbb{G}$ has algebraic length 1,
and $\mathbb{G}$ with unions of pairs of $\mathrm{Aut}(\mathbb{G})$-orbits does not pp-construct $\mathbb{K}_3$
$\Rightarrow \mathbb{G}/\mathrm{Aut}(\mathbb{G})$ contains a loop.

Theorem [Brunar, Kozik, N., Pinsker]

$\mathbb{G}$ smooth ω-categorical digraph, $\mathbb{G}$ has algebraic length 1,
and $\mathbb{G}$ with unions of pairs of $\mathrm{Aut}(\mathbb{G})$-orbits does not pp-construct $\mathbb{K}_3$
$\Rightarrow \mathbb{G}/\,\mathrm{Aut}(\mathbb{G})$ contains a loop.

Proof idea: Find a refinement of the orbit equivalence which has better chances of being pp-definable
de-algebraised (mostly combinatorial) arguments

Theorem [Brunar, Kozik, N., Pinsker]

$\mathbb{G}$ smooth ω-categorical digraph, $\mathbb{G}$ has algebraic length 1,
and $\mathbb{G}$ with unions of pairs of $\mathrm{Aut}(\mathbb{G})$-orbits does not pp-construct $\mathbb{K}_3$
$\Rightarrow \mathbb{G}/\mathrm{Aut}(\mathbb{G})$ contains a loop.

Proof idea: Find a refinement of the orbit equivalence which has better chances of being pp-definable
de-algebraised (mostly combinatorial) arguments

Algebraic consequences:
4-ary pseudo-Siggers for structures which pp-define all unions of pairs of orbits of k-tuples for every k

Theorem [Brunar, Kozik, N., Pinsker]

$\mathbb{G}$ smooth ω-categorical digraph, $\mathbb{G}$ has algebraic length 1,
and $\mathbb{G}$ with unions of pairs of $\mathrm{Aut}(\mathbb{G})$-orbits does not pp-construct $\mathbb{K}_3$
$\Rightarrow \mathbb{G}/\mathrm{Aut}(\mathbb{G})$ contains a loop.

Proof idea: Find a refinement of the orbit equivalence which has better chances of being pp-definable
de-algebraised (mostly combinatorial) arguments

Algebraic consequences:
4-ary pseudo-Siggers for structures which pp-define all unions of pairs of orbits of k-tuples for every k

Hope: Techniques might generalise further

- Further de-algebraising of the finite proofs

- Further de-algebraising of the finite proofs
- New approaches to loop lemmata:
  - Using topology to reprove Hell-Nešetřil (Meyer, Opršal, 2025)
    $\rightsquigarrow$ another kind of algebra

- Further de-algebraising of the finite proofs
- New approaches to loop lemmata:
  - Using topology to reprove Hell-Nešetřil (Meyer, Opršal, 2025) $\rightsquigarrow$ another kind of algebra
  - Connection between identities and PCSPs (Mottet, 2025)

Figure: Thank you for your attention!