Commutative, Idempotent Groupoids And The Constraint Satisfaction Problem

David Failing

July 5, 2013

What Are Constraint Satisfaction Problems?

Informally, a Constraint Satisfaction Problem (CSP) consists of a finite set of variables, ranging over some finite domain of values, and a set of constraints which restrict the values of the variables. The CSP asks whether there is an assignment of values to the variables such that all constraints are satisfied.

Examples

- Graph k-colorability.
- Solvability of systems of linear equations over a finite field.

What Are Constraint Satisfaction Problems?

Informally, a Constraint Satisfaction Problem (CSP) consists of a finite set of variables, ranging over some finite domain of values, and a set of constraints which restrict the values of the variables. The CSP asks whether there is an assignment of values to the variables such that all constraints are satisfied.

Examples

- Graph k-colorability.
- Solvability of systems of linear equations over a finite field.

What Are Constraint Satisfaction Problems?

Informally, a Constraint Satisfaction Problem (CSP) consists of a finite set of variables, ranging over some finite domain of values, and a set of constraints which restrict the values of the variables. The CSP asks whether there is an assignment of values to the variables such that all constraints are satisfied.

Examples

- Graph k-colorability.
- Solvability of systems of linear equations over a finite field.

Single-Sorted CSP

Definition

For any finite set A, and any set Γ of relations over A, **CSP**(Γ) is the combinatorial decision problem:

INSTANCE: A triple $\mathcal{R} = (V, A, \mathcal{C})$ where:

- V a finite set of variables
- $\mathcal{C} = \{(S_i, R_i) \mid i = 1, \ldots, n\}$ a set of constraints, with each S_i a tuple of variables, and each R_i an element of Γ which indicates the allowed simultaneous values for variables in S_i

QUESTION: Does $\mathcal{R}$ have a solution? That is, is there a map $f : V \to A$ such that for all i, $f(S_i) \in R_i$?

Single-Sorted CSP

Definition

For any finite set A, and any set Γ of relations over A, **CSP**(Γ) is the combinatorial decision problem:

INSTANCE: A triple $\mathcal{R} = (V, A, \mathcal{C})$ where:

- V a finite set of variables
- $\mathcal{C} = \{(S_i, R_i) \mid i = 1, \ldots, n\}$ a set of constraints, with each S_i a tuple of variables, and each R_i an element of Γ which indicates the allowed simultaneous values for variables in S_i

QUESTION: Does $\mathcal{R}$ have a solution? That is, is there a map $f : V \to A$ such that for all i, $f(S_i) \in R_i$?

Single-Sorted CSP

Definition

For any finite set A, and any set Γ of relations over A, **CSP**(Γ) is the combinatorial decision problem:

INSTANCE: A triple $\mathcal{R} = (V, A, \mathcal{C})$ where:

- V a finite set of variables
- $\mathcal{C} = \{(S_i, R_i) \mid i = 1, \ldots, n\}$ a set of constraints, with each S_i a tuple of variables, and each R_i an element of Γ which indicates the allowed simultaneous values for variables in S_i

QUESTION: Does $\mathcal{R}$ have a solution? That is, is there a map $f : V \to A$ such that for all i, $f(S_i) \in R_i$?

Single-Sorted CSP

Definition

For any finite set A, and any set Γ of relations over A, **CSP**(Γ) is the combinatorial decision problem:

INSTANCE: A triple $\mathcal{R} = (V, A, \mathcal{C})$ where:

- V a finite set of variables
- $\mathcal{C} = \{(S_i, R_i) \mid i = 1, \ldots, n\}$ a set of constraints, with each S_i a tuple of variables, and each R_i an element of Γ which indicates the allowed simultaneous values for variables in S_i

QUESTION: Does $\mathcal{R}$ have a solution? That is, is there a map $f : V \to A$ such that for all i, $f(S_i) \in R_i$?

Single-Sorted CSP

Definition

For any finite set A, and any set Γ of relations over A, **CSP**(Γ) is the combinatorial decision problem:

INSTANCE: A triple $\mathcal{R} = (V, A, \mathcal{C})$ where:

- V a finite set of variables
- $\mathcal{C} = \{(S_i, R_i) \mid i = 1, \ldots, n\}$ a set of constraints, with each S_i a tuple of variables, and each R_i an element of Γ which indicates the allowed simultaneous values for variables in S_i

QUESTION: Does $\mathcal{R}$ have a solution? That is, is there a map $f\colon V \to A$ such that for all i, $f(S_i) \in R_i$?

Many-Sorted CSP

Definition

An instance of the many-sorted CSP is a quadruple $\mathcal{R} = (V, \mathcal{A}, \delta, \mathcal{C})$ in which:

- V is a finite set of variables,
- $\mathcal{A} = \{A_i \mid i \in I\}$ is a collection of finite sets of values,
- $\delta\colon V \to I$ is called the domain function,
- $\mathcal{C} = \{(S_i, R_i) \mid i = 1, \ldots, n\}$ is a set of constraints. For $1 \le i \le n$, $S_i = (v_1, \ldots, v_{m_i})$ is an m_i-tuple of variables, and each R_i is an m_i-ary relation over $\mathcal{A}$ with signature $(\delta(v_1), \ldots, \delta(v_{m_i}))$ which indicates the allowed simultaneous values for variables in S_i.

Question: Does $\mathcal{R}$ have a solution? That is, is there a map $f\colon V \to \bigcup_{i \in I} A_i$ such that for each $v \in V$, $f(v) \in A_{\delta(v)}$, and for $1 \le i \le n$, $f(S_i) \in R_i$?

Many-Sorted CSP

Definition

An instance of the many-sorted CSP is a quadruple $\mathcal{R} = (V, \mathcal{A}, \delta, \mathcal{C})$ in which:

- V is a finite set of variables,
- $\mathcal{A} = \{A_i \mid i \in I\}$ is a collection of finite sets of values,
- $\delta\colon V \to I$ is called the domain function,
- $\mathcal{C} = \{(S_i, R_i) \mid i = 1, \ldots, n\}$ is a set of constraints. For $1 \leq i \leq n$, $S_i = (v_1, \ldots, v_{m_i})$ is an m_i-tuple of variables, and each R_i is an m_i-ary relation over $\mathcal{A}$ with signature $(\delta(v_1), \ldots, \delta(v_{m_i}))$ which indicates the allowed simultaneous values for variables in S_i.

Question: Does $\mathcal{R}$ have a solution? That is, is there a map $f\colon V \to \bigcup_{i \in I} A_i$ such that for each $v \in V$, $f(v) \in A_{\delta(v)}$, and for $1 \leq i \leq n$, $f(S_i) \in R_i$?

Many-Sorted CSP

Definition

An instance of the many-sorted CSP is a quadruple $\mathcal{R} = (V, \mathcal{A}, \delta, \mathcal{C})$ in which:

- V is a finite set of variables,
- $\mathcal{A} = \{A_i \mid i \in I\}$ is a collection of finite sets of values,
- $\delta \colon V \to I$ is called the domain function,
- $\mathcal{C} = \{(S_i, R_i) \mid i = 1, \ldots, n\}$ is a set of constraints. For $1 \leq i \leq n$, $S_i = (v_1, \ldots, v_{m_i})$ is an m_i-tuple of variables, and each R_i is an m_i-ary relation over $\mathcal{A}$ with signature $(\delta(v_1), \ldots, \delta(v_{m_i}))$ which indicates the allowed simultaneous values for variables in S_i.

Question: Does $\mathcal{R}$ have a solution? That is, is there a map $f \colon V \to \bigcup_{i \in I} A_i$ such that for each $v \in V$, $f(v) \in A_{\delta(v)}$, and for $1 \leq i \leq n$, $f(S_i) \in R_i$?

Many-Sorted CSP

Definition

An instance of the many-sorted CSP is a quadruple $\mathcal{R} = (V, \mathcal{A}, \delta, \mathcal{C})$ in which:

- V is a finite set of variables,
- $\mathcal{A} = \{A_i \mid i \in I\}$ is a collection of finite sets of values,
- $\delta \colon V \to I$ is called the domain function,
- $\mathcal{C} = \{(S_i, R_i) \mid i = 1, \ldots, n\}$ is a set of constraints. For $1 \leq i \leq n$, $S_i = (v_1, \ldots, v_{m_i})$ is an m_i-tuple of variables, and each R_i is an m_i-ary relation over $\mathcal{A}$ with signature $(\delta(v_1), \ldots, \delta(v_{m_i}))$ which indicates the allowed simultaneous values for variables in S_i.

Question: Does $\mathcal{R}$ have a solution? That is, is there a map $f \colon V \to \bigcup_{i \in I} A_i$ such that for each $v \in V$, $f(v) \in A_{\delta(v)}$, and for $1 \leq i \leq n$, $f(S_i) \in R_i$?

Observation

CSP(Γ) is in NP.

Definition

If each instance of **CSP**(Γ) is answerable (yes/no) in polynomial time, we say that **CSP**(Γ) is tractable, or that Γ is a tractable set of relations. Γ is NP-complete if there is some finite $\Delta \subseteq \Gamma$ for which $CSP(\Delta)$ is NP-complete.

Examples

- For k-colorability, if $k = 2$ we can use Breadth First Search to produce a coloring (or show none exists) in polynomial time. For $k \geq 3$, the problem is known to be NP-complete.
- For systems of linear equations over a finite field, Gaussian elimination will find a solution, if it exists, in polynomial time.

Observation

CSP(Γ) is in NP.

Definition

If each instance of **CSP**(Γ) is answerable (yes/no) in polynomial time, we say that **CSP**(Γ) is tractable, or that Γ is a tractable set of relations. Γ is NP-complete if there is some finite $\Delta \subseteq \Gamma$ for which $CSP(\Delta)$ is NP-complete.

Examples

- For k-colorability, if $k = 2$ we can use Breadth First Search to produce a coloring (or show none exists) in polynomial time. For $k \geq 3$, the problem is known to be NP-complete.
- For systems of linear equations over a finite field, Gaussian elimination will find a solution, if it exists, in polynomial time.

Observation

CSP(Γ) is in NP.

Definition

If each instance of **CSP**(Γ) is answerable (yes/no) in polynomial time, we say that **CSP**(Γ) is tractable, or that Γ is a tractable set of relations. Γ is NP-complete if there is some finite $\Delta \subseteq \Gamma$ for which $CSP(\Delta)$ is NP-complete.

Examples

- For k-colorability, if $k = 2$ we can use Breadth First Search to produce a coloring (or show none exists) in polynomial time. For $k \geq 3$, the problem is known to be NP-complete.
- For systems of linear equations over a finite field, Gaussian elimination will find a solution, if it exists, in polynomial time.

Observation

CSP(Γ) is in NP.

Definition

If each instance of **CSP**(Γ) is answerable (yes/no) in polynomial time, we say that **CSP**(Γ) is tractable, or that Γ is a tractable set of relations. Γ is NP-complete if there is some finite $\Delta \subseteq \Gamma$ for which $CSP(\Delta)$ is NP-complete.

Examples

- For k-colorability, if $k = 2$ we can use Breadth First Search to produce a coloring (or show none exists) in polynomial time. For $k \geq 3$, the problem is known to be NP-complete.
- For systems of linear equations over a finite field, Gaussian elimination will find a solution, if it exists, in polynomial time.

Two Problems

Theorem (Bulatov & Jeavons '01)

Every many-sorted CSP can be transformed into a single-sorted CSP which has the same complexity.

CSP Dichotomy Conjecture (Feder & Vardi '98)

Every $\mathbf{CSP}(\Gamma)$ is either tractable, or it is NP-complete.

Problem

Characterize *all* tractable sets of relations.

Two Problems

Theorem (Bulatov & Jeavons '01)

Every many-sorted CSP can be transformed into a single-sorted CSP which has the same complexity.

CSP Dichotomy Conjecture (Feder & Vardi '98)

Every **CSP**(Γ) is either tractable, or it is NP-complete.

Problem

Characterize *all* tractable sets of relations.

Two Problems

Theorem (Bulatov & Jeavons '01)

Every many-sorted CSP can be transformed into a single-sorted CSP which has the same complexity.

CSP Dichotomy Conjecture (Feder & Vardi '98)

Every **CSP**(Γ) is either tractable, or it is NP-complete.

Problem

Characterize *all* tractable sets of relations.

The Universal Algebraic Approach

Definition

We say that an m-ary operation $f\colon A^m \to A$ preserves an n-ary relation R over A (or that R is invariant under f) if

$$\overline{a}_1, \ldots, \overline{a}_m \in R \Rightarrow f(\overline{a}_1, \ldots, \overline{a}_m) \in R$$

For Γ a set of relations over A and $\mathcal{F}$ a set of operations on A:

$\mathrm{Pol}(\Gamma) := \{f \mid f \text{ preserves every } R \in \Gamma\}$,
the clone of polymorphisms of Γ.

$\mathrm{Inv}(\mathcal{F}) := \{R \mid R \text{ is invariant under every } f \in \mathcal{F}\}$,
the relations invariant under $\mathcal{F}$.

The Universal Algebraic Approach

Definition

We say that an m-ary operation $f\colon A^m \to A$ preserves an n-ary relation R over A (or that R is invariant under f) if

$$\overline{a}_1, \ldots, \overline{a}_m \in R \Rightarrow f(\overline{a}_1, \ldots, \overline{a}_m) \in R$$

For Γ a set of relations over A and $\mathcal{F}$ a set of operations on A:

$\mathrm{Pol}(\Gamma) := \{f \mid f \text{ preserves every } R \in \Gamma\}$,
the clone of polymorphisms of Γ.

$\mathrm{Inv}(\mathcal{F}) := \{R \mid R \text{ is invariant under every } f \in \mathcal{F}\}$,
the relations invariant under $\mathcal{F}$.

The Universal Algebraic Approach

Definition

We say that an m-ary operation $f\colon A^m \to A$ preserves an n-ary relation R over A (or that R is invariant under f) if

$$\overline{a}_1, \ldots, \overline{a}_m \in R \Rightarrow f(\overline{a}_1, \ldots, \overline{a}_m) \in R$$

For Γ a set of relations over A and $\mathcal{F}$ a set of operations on A:

$\mathrm{Pol}(\Gamma) := \{f \mid f \text{ preserves every } R \in \Gamma\}$,
the clone of polymorphisms of Γ.

$\mathrm{Inv}(\mathcal{F}) := \{R \mid R \text{ is invariant under every } f \in \mathcal{F}\}$,
the relations invariant under $\mathcal{F}$.

The Universal Algebraic Approach

Definition

We say that an m-ary operation $f\colon A^m \to A$ preserves an n-ary relation R over A (or that R is invariant under f) if

$$\bar{a}_1, \ldots, \bar{a}_m \in R \Rightarrow f(\bar{a}_1, \ldots, \bar{a}_m) \in R$$

For Γ a set of relations over A and $\mathcal{F}$ a set of operations on A:

$\mathrm{Pol}(\Gamma) := \{f \mid f \text{ preserves every } R \in \Gamma\}$,
the clone of polymorphisms of Γ.

$\mathrm{Inv}(\mathcal{F}) := \{R \mid R \text{ is invariant under every } f \in \mathcal{F}\}$,
the relations invariant under $\mathcal{F}$.

The Universal Algebraic Approach

Definition

An algebra is a pair $\mathbf{A} = \langle A, \mathcal{F} \rangle$, where A is a nonempty set, and $\mathcal{F}$ is a set of operations on A.

Observation

To every set of relations Γ over a finite set A, we can associate the algebra $\mathbf{A} = \langle A, \mathrm{Pol}(\Gamma) \rangle$. Likewise, to every finite algebra $\mathbf{A} = \langle A, \mathcal{F} \rangle$, we can associate the set of relations $\mathrm{Inv}(\mathcal{F})$.

Definition

We say an algebra $\mathbf{A} = \langle A, \mathcal{F} \rangle$ is tractable if $\mathrm{Inv}(\mathcal{F})$ is a tractable set of relations. Similarly, $\mathbf{A}$ may be NP-complete. We can consider only idempotent algebras. $(\forall f \in \mathcal{F},\ f(x, x, \ldots, x) \approx x)$

The Universal Algebraic Approach

Definition

An algebra is a pair $\mathbf{A} = \langle A, \mathcal{F} \rangle$, where A is a nonempty set, and $\mathcal{F}$ is a set of operations on A.

Observation

To every set of relations Γ over a finite set A, we can associate the algebra $\mathbf{A} = \langle A, \mathrm{Pol}(\Gamma) \rangle$. Likewise, to every finite algebra $\mathbf{A} = \langle A, \mathcal{F} \rangle$, we can associate the set of relations $\mathrm{Inv}(\mathcal{F})$.

Definition

We say an algebra $\mathbf{A} = \langle A, \mathcal{F} \rangle$ is tractable if $\mathrm{Inv}(\mathcal{F})$ is a tractable set of relations. Similarly, $\mathbf{A}$ may be NP-complete. We can consider only idempotent algebras. $(\forall f \in \mathcal{F},\ f(x, x, \ldots, x) \approx x)$

The Universal Algebraic Approach

Definition

An algebra is a pair $\mathbf{A} = \langle A, \mathcal{F} \rangle$, where A is a nonempty set, and $\mathcal{F}$ is a set of operations on A.

Observation

To every set of relations Γ over a finite set A, we can associate the algebra $\mathbf{A} = \langle A, \text{Pol}(\Gamma) \rangle$. Likewise, to every finite algebra $\mathbf{A} = \langle A, \mathcal{F} \rangle$, we can associate the set of relations $\text{Inv}(\mathcal{F})$.

Definition

We say an algebra $\mathbf{A} = \langle A, \mathcal{F} \rangle$ is tractable if $\text{Inv}(\mathcal{F})$ is a tractable set of relations. Similarly, $\mathbf{A}$ may be NP-complete. We can consider only idempotent algebras. $(\forall f \in \mathcal{F},\ f(x, x, \ldots, x) \approx x)$

Classes of Algebras

Definition

A variety $\mathcal{V}$ is a class of algebras which is closed under homomorphic images, subalgebras, and direct products. We say that a variety is tractable if every one of its finite members is tractable.

Definition

An algebra is congruence meet-semidistributive (SD($\wedge$)) if its congruence lattice satisfies

$$(x \wedge y \approx x \wedge z) \Rightarrow (x \wedge (y \vee z) \approx x \wedge y).$$

A class $\mathcal{K}$ of algebras is SD($\wedge$) if every algebra in $\mathcal{K}$ is SD($\wedge$).

Definition

A variety $\mathcal{V}$ is a class of algebras which is closed under homomorphic images, subalgebras, and direct products. We say that a variety is tractable if every one of its finite members is tractable.

Definition

An algebra is congruence meet-semidistributive (SD($\wedge$)) if its congruence lattice satisfies

$$(x \wedge y \approx x \wedge z) \Rightarrow (x \wedge (y \vee z) \approx x \wedge y).$$

A class $\mathcal{K}$ of algebras is SD($\wedge$) if every algebra in $\mathcal{K}$ is SD($\wedge$).

Term Conditions

Definition

A ternary operation q is Maltsev if it satisfies

$$q(x, y, y) \approx q(y, y, x) \approx x.$$

Example

For $\mathbf{G} \approx \langle G, \cdot, ^{-1}, e \rangle$ a group, $q(x, y, z) \approx x \cdot y^{-1} \cdot z$.

Example

For $\mathbf{Q} = \langle Q, \cdot, /, \backslash \rangle$ a quasigroup, $q(x, y, z) = (x/(y\backslash y)) \cdot (y \backslash z)$.

Term Conditions

Definition

A ternary operation q is Maltsev if it satisfies

$$q(x, y, y) \approx q(y, y, x) \approx x.$$

Example

For $\mathbf{G} \approx \langle G, \cdot, ^{-1}, e \rangle$ a group, $q(x, y, z) \approx x \cdot y^{-1} \cdot z$.

Example

For $\mathbf{Q} = \langle Q, \cdot, /, \backslash \rangle$ a quasigroup, $q(x, y, z) = (x/(y\backslash y)) \cdot (y\backslash z)$.

Term Conditions

Definition

A ternary operation q is Maltsev if it satisfies

$$q(x, y, y) \approx q(y, y, x) \approx x.$$

Example

For $\mathbf{G} \approx \langle G, \cdot, ^{-1}, e \rangle$ a group, $q(x, y, z) \approx x \cdot y^{-1} \cdot z$.

Example

For $\mathbf{Q} = \langle Q, \cdot, /, \backslash \rangle$ a quasigroup, $q(x, y, z) = (x/(y\backslash y)) \cdot (y\backslash z)$.

Term Conditions

Definition

A k-ary weak near-unanimity operation on A is an idempotent operation that satisfies the identities

$$f(y, x, \ldots, x) \approx f(x, y, \ldots, x) \approx \cdots \approx f(x, x, \ldots, x, y).$$

A k-ary near-unanimity operation is a weak near-unanimity operation that satisfies the identity $f(y, x, \ldots, x) \approx x$.

Term Conditions

Definition

For $k \geq 2$, a k-edge operation on a set A is a $(k+1)$-ary operation, f, on A satisfying the k identities:

$$
\begin{aligned}
f(x,x,y,y,y,\dots,y,y) &\approx y \\
f(x,y,x,y,y,\dots,y,y) &\approx y \\
f(y,y,y,x,y,\dots,y,y) &\approx y \\
f(y,y,y,y,x,\dots,y,y) &\approx y \\
&\vdots \\
f(y,y,y,y,y,\dots,x,y) &\approx y \\
f(y,y,y,y,y,\dots,y,x) &\approx y
\end{aligned}
$$

Two Main Algorithms

Theorem (Barto & Kozik '09)

Any finite algebra which lies in a congruence meet-semidistributive variety is tractable.

Theorem (IMMVW '10)

If $\mathrm{Clo}(\mathcal{F})$ *contains an edge term, then* $\mathbf{A} = \langle A, \mathcal{F} \rangle$ *is tractable. Every Maltsev term and NU term gives rise to an edge term.*

Two Main Algorithms

Theorem (Barto & Kozik '09)

Any finite algebra which lies in a congruence meet-semidistributive variety is tractable.

Theorem (IMMVW '10)

If $\mathrm{Clo}(\mathcal{F})$ *contains an edge term, then* $\mathbf{A} = \langle A, \mathcal{F} \rangle$ *is tractable. Every Maltsev term and NU term gives rise to an edge term.*

Restating The Problem

Theorem (Bulatov, Jeavons, Krokhin '05; Maroti & McKenzie '08)

Let $\mathbf{A} = \langle A, \mathcal{F} \rangle$ *be a finite algebra. If* $\mathrm{Clo}(\mathcal{F})$ *contains no weak near-unanimity operation, then* $\mathbf{A}$ *is NP-complete.*

Algebraic Dichotomy Conjecture

If $\mathrm{Clo}(\mathcal{F})$ contains a WNU, then $\mathbf{A}$ is tractable.

Restating The Problem

Theorem (Bulatov, Jeavons, Krokhin '05; Maroti & McKenzie '08)

Let $\mathbf{A} = \langle A, \mathcal{F} \rangle$ be a finite algebra. If $\mathrm{Clo}(\mathcal{F})$ *contains no weak near-unanimity operation, then* $\mathbf{A}$ *is NP-complete.*

Algebraic Dichotomy Conjecture

If $\mathrm{Clo}(\mathcal{F})$ contains a WNU, then $\mathbf{A}$ is tractable.

Approaching The Dichotomy Conjecture

- A binary term is a WNU iff it is commutative and idempotent.
- Neither alone is sufficient for tractability.
- The variety of semilattices (associative, idempotent, commutative groupoids) is SD($\wedge$), and tractable.
- We studied commutative, idempotent groupoids satisfying identities strictly weaker than associativity. Why?
- If the Algebraic Dichotomy Conjecture is true, any weakening of associativity (with C,I) should also suffice for tractability.

Approaching The Dichotomy Conjecture

- A binary term is a WNU iff it is commutative and idempotent.
- Neither alone is sufficient for tractability.
- The variety of semilattices (associative, idempotent, commutative groupoids) is SD($\wedge$), and tractable.
- We studied commutative, idempotent groupoids satisfying identities strictly weaker than associativity. Why?
- If the Algebraic Dichotomy Conjecture is true, any weakening of associativity (with C,I) should also suffice for tractability.

Approaching The Dichotomy Conjecture

- A binary term is a WNU iff it is commutative and idempotent.
- Neither alone is sufficient for tractability.
- The variety of semilattices (associative, idempotent, commutative groupoids) is SD($\wedge$), and tractable.
- We studied commutative, idempotent groupoids satisfying identities strictly weaker than associativity. Why?
- If the Algebraic Dichotomy Conjecture is true, any weakening of associativity (with C,I) should also suffice for tractability.

Approaching The Dichotomy Conjecture

- A binary term is a WNU iff it is commutative and idempotent.
- Neither alone is sufficient for tractability.
- The variety of semilattices (associative, idempotent, commutative groupoids) is SD($\wedge$), and tractable.
- We studied commutative, idempotent groupoids satisfying identities strictly weaker than associativity. Why?
- If the Algebraic Dichotomy Conjecture is true, any weakening of associativity (with C,I) should also suffice for tractability.

Approaching The Dichotomy Conjecture

- A binary term is a WNU iff it is commutative and idempotent.
- Neither alone is sufficient for tractability.
- The variety of semilattices (associative, idempotent, commutative groupoids) is SD($\wedge$), and tractable.
- We studied commutative, idempotent groupoids satisfying identities strictly weaker than associativity. Why?
- If the Algebraic Dichotomy Conjecture is true, any weakening of associativity (with C,I) should also suffice for tractability.

First Generalization

Definition

Let $\mathbf{A} = \langle A, \cdot \rangle$ be a groupoid. We call $\mathbf{A}$ a CI-groupoid if $\cdot$ is both commutative and idempotent. Usually, we write xy for $x \cdot y$.

Definition

An identity $p \approx q$ is of Bol-Moufang type if (i) the only operation in p, q is $\cdot$, (ii) the same three variables appear on both sides, in the same order, (iii) one of the variables appears twice, (iv) the remaining two variables appear only once.

Example

The Moufang Law $x(y(zy)) \approx ((xy)z)y$ is an identity of Bol-Moufang type.

First Generalization

Definition

Let $\mathbf{A} = \langle A, \cdot \rangle$ be a groupoid. We call $\mathbf{A}$ a CI-groupoid if $\cdot$ is both commutative and idempotent. Usually, we write xy for $x \cdot y$.

Definition

An identity $p \approx q$ is of Bol-Moufang type if (i) the only operation in p, q is $\cdot$, (ii) the same three variables appear on both sides, in the same order, (iii) one of the variables appears twice, (iv) the remaining two variables appear only once.

Example

The Moufang Law $x(y(zy)) \approx ((xy)z)y$ is an identity of Bol-Moufang type.

First Generalization

Definition

Let $\mathbf{A} = \langle A, \cdot \rangle$ be a groupoid. We call $\mathbf{A}$ a CI-groupoid if $\cdot$ is both commutative and idempotent. Usually, we write xy for $x \cdot y$.

Definition

An identity $p \approx q$ is of Bol-Moufang type if (i) the only operation in p, q is $\cdot$, (ii) the same three variables appear on both sides, in the same order, (iii) one of the variables appears twice, (iv) the remaining two variables appear only once.

Example

The Moufang Law $x(y(zy)) \approx ((xy)z)y$ is an identity of Bol-Moufang type.

Identities of Bol-Moufang Type (Phillips and Vojtěchovský)

A	$xxyz$
B	$xyxz$
C	$xyyz$
D	$xyzx$
E	$xyzy$
F	$xyzz$

1	$o(o(oo))$
2	$o((oo)o)$
3	$(oo)(oo)$
4	$(o(oo))o$
5	$((oo)o)o$

- Representable as Xij, the identity with:
  - variable order X
  - LHS bracketed by i, and RHS bracketed by j.
- $x(y(zy)) \approx ((xy)z)y$ is represented as $E15$.
- There are $6 * (4 + 3 + 2 + 1) = 60$ nontrivial such identities.

Identities of Bol-Moufang Type (Phillips and Vojtěchovský)

A	$xxyz$
B	$xyxz$
C	$xyyz$
D	$xyzx$
E	$xyzy$
F	$xyzz$

1	$o(o(oo))$
2	$o((oo)o)$
3	$(oo)(oo)$
4	$(o(oo))o$
5	$((oo)o)o$

- Representable as Xij, the identity with:
  - variable order X
  - LHS bracketed by i, and RHS bracketed by j.
- $x(y(zy)) \approx ((xy)z)y$ is represented as $E15$.
- There are $6 * (4 + 3 + 2 + 1) = 60$ nontrivial such identities.

Identities of Bol-Moufang Type (Phillips and Vojtěchovský)

A	$xxyz$
B	$xyxz$
C	$xyyz$
D	$xyzx$
E	$xyzy$
F	$xyzz$

1	$o(o(oo))$
2	$o((oo)o)$
3	$(oo)(oo)$
4	$(o(oo))o$
5	$((oo)o)o$

- Representable as Xij, the identity with:
  - variable order X
  - LHS bracketed by i, and RHS bracketed by j.
- $x(y(zy)) \approx ((xy)z)y$ is represented as $E15$.
- There are $6 * (4 + 3 + 2 + 1) = 60$ nontrivial such identities.

Varieties of Bol-Moufang Type

- A variety of ________________ is of Bol-Moufang type if it is axiomatized by one additional identity of Bol-Moufang type.
- Two identities of BM type are equivalent if they axiomatize the same variety of ________________ of B-M type.
- Phillips and Vojtěchovský showed there are 26 varieties of quasigroups, and 14 varieties of loops of B-M type.
- Q: How many varieties of CI-Groupoids of B-M type are there? What structure do they have?
  - Checking pairwise equivalence by hand would take too long!

Varieties of Bol-Moufang Type

- A variety of ________________ is of Bol-Moufang type if it is axiomatized by one additional identity of Bol-Moufang type.
- Two identities of BM type are equivalent if they axiomatize the same variety of ________________ of B-M type.
- Phillips and Vojtěchovský showed there are 26 varieties of quasigroups, and 14 varieties of loops of B-M type.
- Q: How many varieties of CI-Groupoids of B-M type are there? What structure do they have?
  - Checking pairwise equivalence by hand would take too long!

Varieties of Bol-Moufang Type

- A variety of ________________ is of Bol-Moufang type if it is axiomatized by one additional identity of Bol-Moufang type.
- Two identities of BM type are equivalent if they axiomatize the same variety of ________________ of B-M type.
- Phillips and Vojtěchovský showed there are 26 varieties of quasigroups, and 14 varieties of loops of B-M type.
- Q: How many varieties of CI-Groupoids of B-M type are there? What structure do they have?
  - Checking pairwise equivalence by hand would take too long!

Varieties of Bol-Moufang Type

- A variety of ________________ is of Bol-Moufang type if it is axiomatized by one additional identity of Bol-Moufang type.
- Two identities of BM type are equivalent if they axiomatize the same variety of ________________ of B-M type.
- Phillips and Vojtěchovský showed there are 26 varieties of quasigroups, and 14 varieties of loops of B-M type.
- Q: How many varieties of CI-Groupoids of B-M type are there? What structure do they have?
  - Checking pairwise equivalence by hand would take too long!

- **Prover9** is an automated theorem prover for first-order and equational logic.
- **Mace4** searches for finite models and counterexamples.
- **Input:** a set of assumptions and one or more goals.
- **Output:** either proofs of the goals, or counterexample(s) where the assumptions are true but the goals are false.

Prover9 / Mace4

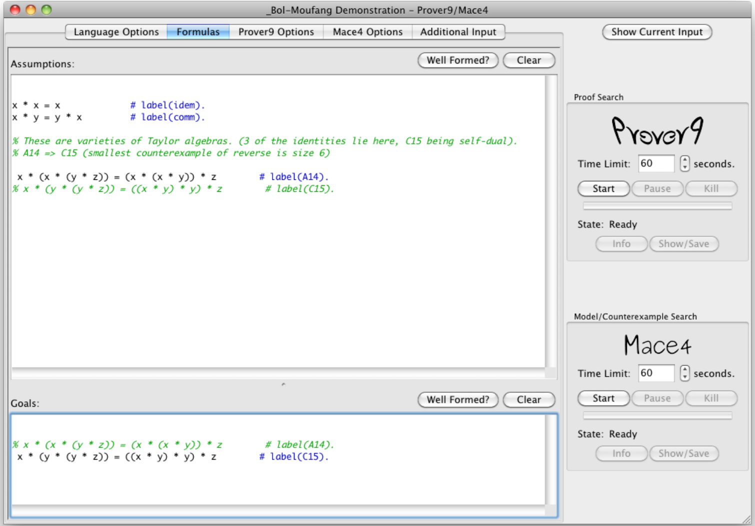
- **Prover9** is an automated theorem prover for first-order and equational logic.
- **Mace4** searches for finite models and counterexamples.
- **Input:** a set of assumptions and one or more goals.
- **Output:** either proofs of the goals, or counterexample(s) where the assumptions are true but the goals are false.

- **Prover9** is an automated theorem prover for first-order and equational logic.
- **Mace4** searches for finite models and counterexamples.
- **Input:** a set of assumptions and one or more goals.
- **Output:** either proofs of the goals, or counterexample(s) where the assumptions are true but the goals are false.

Prover9 / Mace4

- **Prover9** is an automated theorem prover for first-order and equational logic.
- **Mace4** searches for finite models and counterexamples.
- **Input:** a set of assumptions and one or more goals.
- **Output:** either proofs of the goals, or counterexample(s) where the assumptions are true but the goals are false.

Prover9 / Mace4 Demonstration



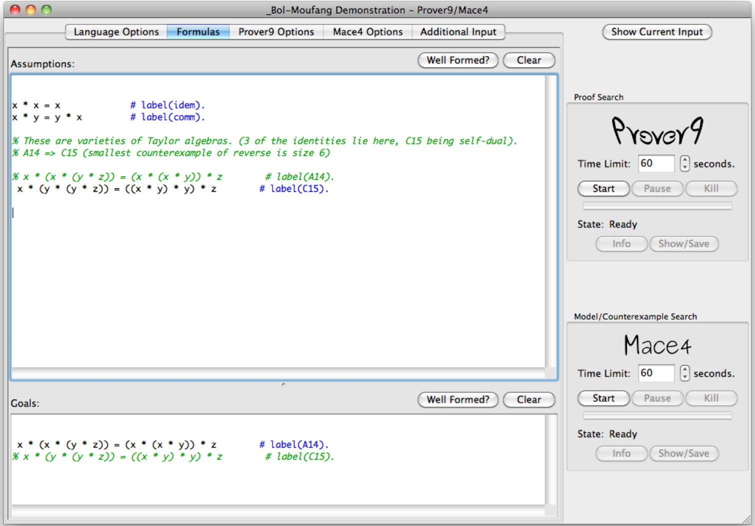
Prover9 / Mace4 Demonstration

Prover9 Proof

Save as... Reformat ... Close

```
============================== prooftrans ============================
Prover9 (32) version Dec-2007, Dec 2007.
Process 7640 was started by David on David-Failings-MacBook-Pro-2.local,
Fri Jun 28 08:53:25 2013
The command was "/Applications/Prover9-Mace4-v05B.app/Contents/Resources/bin-mac-intel/prover9".
============================== end of head ===========================

============================== end of input ==========================

============================== PROOF =================================

% -------- Comments from original proof --------
% Proof 1 at 0.01 (+ 0.01) seconds: C15.
% Length of proof is 10.
% Level of proof is 4.
% Maximum clause weight is 15.
% Given clauses 10.

1 x * (y * (y * z)) = ((x * y) * y) * z # label(C15) # label(non_clause) # label(goal).  [goal].
3 x * y = y * x # label(comm).  [assumption].
4 x * (x * (y * z)) = (x * (x * y)) * z # label(A14).  [assumption].
5 (x * (x * y)) * z = x * (x * (y * z)).  [copy(4),flip(a)].
6 ((c1 * c2) * c2) * c3 != c1 * (c2 * (c2 * c3)) # label(C15) # answer(C15).  [deny(1)].
7 c3 * (c2 * (c1 * c2)) != c1 * (c2 * (c2 * c3)) # answer(C15).  [copy(6),rewrite([3(5),3(7)])].
12 x * (y * (y * z)) = y * (y * (z * x)).  [para(5(a,1),3(a,1)),flip(a)].
39 x * (y * (z * y)) = y * (y * (z * x)).  [para(3(a,1),12(a,1,2,2))].
51 c3 * (c2 * (c1 * c2)) != c2 * (c2 * (c1 * c3)) # answer(C15).  [para(12(a,1),7(a,2)),rewrite([3(12)])].
52 $F # answer(C15).  [resolve(51,a,39,a)].

============================== end of proof ==========================
```

Prover9 / Mace4 Demonstration



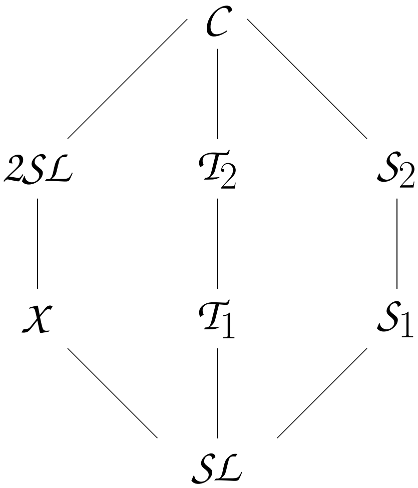
Prover9 / Mace4 Demonstration

Mace4 Model

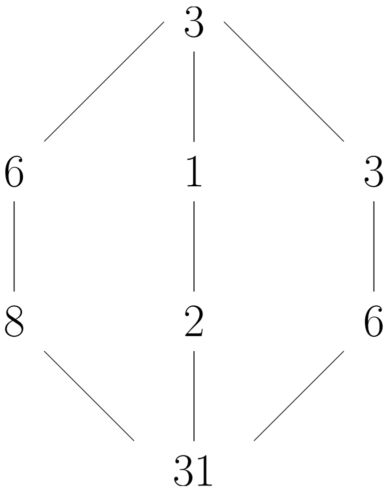
Save as... Reformat ... Close

```
interpretation( 6, [number = 1,seconds = 0], [
    function(*(_,_), [
        0,0,0,4,5,4,
        0,1,3,2,5,4,
        0,3,2,1,5,4,
        4,2,1,3,0,5,
        5,5,5,0,4,0,
        4,4,4,5,0,5]),
    function(c1, [0]),
    function(c2, [1]),
    function(c3, [2])]).
```

The 8 Varieties of CI-Groupoids of Bol-Moufang Type



The 8 Varieties of CI-Groupoids of Bol-Moufang Type



The 8 Varieties of CI-Groupoids of Bol-Moufang Type

$\mathbb{C}$	All CI-groupoids
$2\mathbb{SL}$	$x(xy) \approx xy$
$\mathbb{X}$	$A24$: $x((xy)z) \approx (x(xy))z$
$\mathbb{SL}$	Semilattices
$\mathbb{T}_2$	$C15$: $x(y(yz)) \approx ((xy)y)z$
$\mathbb{T}_1$	$A14$: $x(x(yz)) \approx (x(xy))z$
$\mathbb{S}_2$	$B12$: $x(y(xz)) \approx x((yx)z)$
$\mathbb{S}_1$	$B13$: $x(y(xz)) \approx (xy)(xz)$

SD(∧) Varieties

Theorem (Kearnes & Kiss)

Let $\mathcal{V}$ be a variety of algebras. The following are equivalent:

- *$\mathcal{V}$ is congruence meet-semidistributive*
- *$\mathcal{V}$ satisfies a family of idempotent Maltsev conditions that, considered together, fail in any nontrivial variety of modules.*

We can restrict our attention to modules over unital rings satisfying the implication $rx \approx 0 \Rightarrow r = 0_R$.

SD(∧) Varieties

Theorem (Kearnes & Kiss)

Let $\mathcal{V}$ be a variety of algebras. The following are equivalent:

- *$\mathcal{V}$ is congruence meet-semidistributive*
- *$\mathcal{V}$ satisfies a family of idempotent Maltsev conditions that, considered together, fail in any nontrivial variety of modules.*

We can restrict our attention to modules over unital rings satisfying the implication $rx \approx 0 \Rightarrow r = 0_R$.

SD(∧) Varieties

Theorem

Five of the varieties of CI-groupoids of Bol-Moufang type are SD(∧), *and thus tractable:* $2\mathcal{SL}$, $\mathcal{X}$, $\mathcal{SL}$, $\mathcal{S}_2$, *and* $\mathcal{S}_1$.

Proof. (For $2\mathcal{SL}$, $\mathcal{X}$, $\mathcal{SL}$).

Use the Kearnes & Kiss result. Let the family of identities be commutativity, idempotence, and the 2-semilattice law. Let $\mathcal{M}$ be a variety of modules. Any binary module term is of the form $x \cdot y = rx + sy$.

- $x \cdot y \approx y \cdot x \Rightarrow r = s$
- $x \cdot x \approx x \Rightarrow r + r = 1_R$
- $x \cdot (x \cdot y) \approx x \cdot y \Rightarrow r^2x + (r^2 - r)y \approx 0 \Rightarrow r^2 = r = 0_R$

Returning to idempotence, $\mathcal{M}$ satisfies

$$x \approx x \cdot x \approx 0_R x + 0_R x \approx 0. \quad \square$$

SD(∧) Varieties

Theorem

Five of the varieties of CI-groupoids of Bol-Moufang type are SD(∧)*, and thus tractable:* $\mathcal{2SL}$, $\mathcal{X}$, $\mathcal{SL}$, $\mathcal{S}_2$, *and* $\mathcal{S}_1$.

Proof. (For $\mathcal{2SL}$, $\mathcal{X}$, $\mathcal{SL}$).

Use the Kearnes & Kiss result. Let the family of identities be commutativity, idempotence, and the 2-semilattice law. Let $\mathcal{M}$ be a variety of modules. Any binary module term is of the form $x \cdot y = rx + sy$.

- $x \cdot y \approx y \cdot x \Rightarrow r = s$
- $x \cdot x \approx x \Rightarrow r + r = 1_R$
- $x \cdot (x \cdot y) \approx x \cdot y \Rightarrow r^2 x + (r^2 - r)y \approx 0 \Rightarrow r^2 = r = 0_R$

Returning to idempotence, $\mathcal{M}$ satisfies

$$x \approx x \cdot x \approx 0_R x + 0_R x \approx 0. \qquad \square$$

Theorem

$\mathcal{T}_1$ (and hence $\mathcal{T}_2$) is not congruence meet-semidistributive.

Proof.

It is enough to produce a nontrivial variety $\mathcal{M}$ of modules, together with a family of Maltsev conditions satisfied in $\mathcal{T}_1$ and $\mathcal{M}$. Let the family be commutativity, idempotence, and $x(x(yz)) \approx (x(xy))z$. Let $\mathcal{M}$ be the variety of modules over $\mathbb{Z}_3$. Define $x \cdot y = 2x + 2y$. Then $\mathcal{M}$ satisfies

- $x \cdot y \approx 2x + 2y \approx 2y + 2x \approx y \cdot x$
- $x \cdot x \approx 2x + 2x \approx (2+2)x \approx x$
- $x \cdot (x \cdot (y \cdot z)) \approx 2x + 4x + 8y + 8z \approx 8x + 4x + 8y + 2z$ $\approx 4x + 8x + 8y + 2z \approx (x \cdot (x \cdot y)) \cdot z$.

□

Theorem

$\mathcal{T}_1$ (and hence $\mathcal{T}_2$) is not congruence meet-semidistributive.

Proof.

It is enough to produce a nontrivial variety $\mathcal{M}$ of modules, together with a family of Maltsev conditions satisfied in $\mathcal{T}_1$ and $\mathcal{M}$. Let the family be commutativity, idempotence, and $x(x(yz)) \approx (x(xy))z$. Let $\mathcal{M}$ be the variety of modules over $\mathbb{Z}_3$. Define $x \cdot y = 2x + 2y$. Then $\mathcal{M}$ satisfies

- $x \cdot y \approx 2x + 2y \approx 2y + 2x \approx y \cdot x$
- $x \cdot x \approx 2x + 2x \approx (2+2)x \approx x$
- $x \cdot (x \cdot (y \cdot z)) \approx 2x + 4x + 8y + 8z \approx 8x + 4x + 8y + 2z$ $\approx 4x + 8x + 8y + 2z \approx (x \cdot (x \cdot y)) \cdot z$.

□

The Variety $\mathcal{T}_1$

Definition

The class of CI-groupoids defined by the additional identity $x(yx) \approx y$ is known as the variety of Steiner quasigroups (squags).

Theorem

$\mathbb{T}_1$ *contains the variety of squags.*

Proof.

For squags, $x(x(yz)) \approx yz \approx (x(xy))z$, so $A14$ holds. □

The Variety $\mathcal{T}_1$

Definition

The class of CI-groupoids defined by the additional identity $x(yx) \approx y$ is known as the variety of Steiner quasigroups (squags).

Theorem

$\mathbb{T}_1$ *contains the variety of squags.*

Proof.

For squags, $x(x(yz)) \approx yz \approx (x(xy))z$, so $A14$ holds. □

The Variety $\mathcal{T}_1$

Definition

The class of CI-groupoids defined by the additional identity $x(yx) \approx y$ is known as the variety of Steiner quasigroups (squags).

Theorem

$\mathbb{T}_1$ contains the variety of squags.

Proof.

For squags, $x(x(yz)) \approx yz \approx (x(xy))z$, so $A14$ holds. □

Płonka Sums

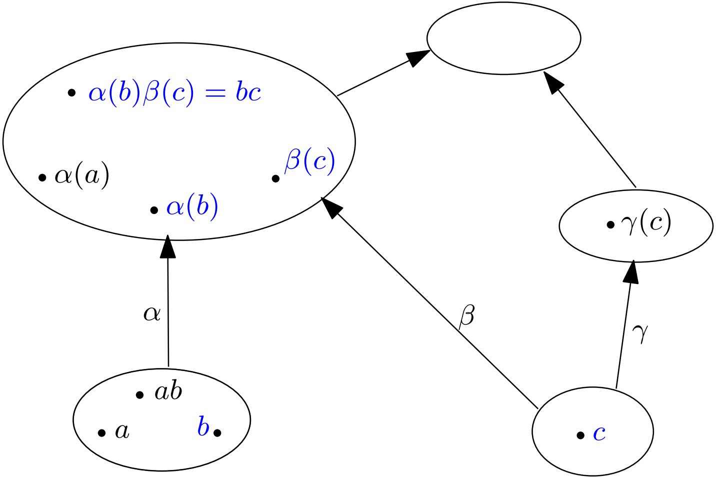
Definition

Let $\mathbf{S} = \langle S, \vee \rangle$ be a semilattice, considered as a category with morphisms $s \to t \Leftrightarrow s \leq t$ in $\mathbf{S}$, V a variety of groupoids considered as a category, and $F\colon S \to V$ a functor. Then the **Płonka sum** over S of the groupoids $\{\mathbf{A}_s = F(s) : s \in S\}$ is the groupoid $\mathbf{A}$ with universe $\dot{\bigcup}_{s \in S} A_s$ and multiplication given by:

$$x_1 \cdot^{\mathbf{A}} x_2 = F_{s_1 s}(x_1) \cdot^{\mathbf{A}_s} F_{s_2 s}(x_2)$$

where $x_i \in \mathbf{A}_{s_i}$, $s = s_1 \vee s_2$, and $F_{s_i s} = F(s_i \to s)$

The Płonka Sum of Groupoids



Płonka's Theorem

Let $\mathcal{V}$ be a variety of groupoids defined by identities $\Sigma \cup \{x \vee y \approx x\}$ for some set Σ of regular identities, and $x \vee y$ a binary term. The following classes of algebras coincide:

1. The class $\mathbf{Pł}(\mathcal{V})$ of Płonka sums of groupoids from $\mathcal{V}$.
2. The variety of groupoids defined by Σ and the identities:

$$x \vee x \approx x \tag{P1}$$

$$(x \vee y) \vee z \approx x \vee (y \vee z) \tag{P2}$$

$$x \vee (y \vee z) \approx x \vee (z \vee y) \tag{P3}$$

$$x \vee (y * z) \approx x \vee y \vee z \tag{P4}$$

$$(x * y) \vee z \approx (x \vee z) * (y \vee z) \tag{P5}$$

Pseudopartition Operations

Definition

We call a binary term $x \vee y$ satisfying (P1)–(P4) in Płonka's Theorem a pseudopartition operation.

Lemma

An algebra $\mathbf{A}$ *possessing a pseudopartition operation has a semilattice replica* $\mathbf{A}/\sigma$, *where* $a \, \sigma \, b \Leftrightarrow (a \vee b = a \text{ and } b \vee a = b)$. $\mathbf{A}$ *also has well-defined maps (which may not be homomorphisms) between its* σ*-classes given by*

$$\phi_{a,b} \colon a/\sigma \to b/\sigma; \, x \mapsto x \vee b.$$

Pseudopartition Operations

Definition

We call a binary term $x \vee y$ satisfying (P1)–(P4) in Płonka's Theorem a pseudopartition operation.

Lemma

An algebra $\mathbf{A}$ *possessing a pseudopartition operation has a semilattice replica* $\mathbf{A}/\sigma$*, where* $a \,\sigma\, b \Leftrightarrow (a \vee b = a \text{ and } b \vee a = b)$*.* $\mathbf{A}$ *also has well-defined maps (which may not be homomorphisms) between its* σ*-classes given by*

$$\phi_{a,b} \colon a/\sigma \to b/\sigma;\ x \mapsto x \vee b.$$

The Variety $\mathcal{T}_2$

Theorem (Main Result)

Let $\mathbf{A}$ *be a finite idempotent algebra with pseudopartition operation* $x \vee y$*, such that every block of its semilattice replica congruence lies in the same tractable variety. Then* $\mathrm{CSP}(\mathbf{A})$ *is tractable.*

Proof.

Idea: Look in the biggest block possible! □

Theorem

$\mathcal{T}_2$ *has a pseudopartition operation* $(x \vee y = y(xy))$*, so it is tractable.*

The Variety $\mathcal{T}_2$

Theorem (Main Result)

Let $\mathbf{A}$ *be a finite idempotent algebra with pseudopartition operation* $x \vee y$*, such that every block of its semilattice replica congruence lies in the same tractable variety. Then* $\mathrm{CSP}(\mathbf{A})$ *is tractable.*

Proof.

Idea: Look in the biggest block possible! □

Theorem

$\mathcal{T}_2$ *has a pseudopartition operation* $(x \vee y = y(xy))$*, so it is tractable.*

The Variety $\mathcal{T}_2$

Theorem (Main Result)

Let $\mathbf{A}$ *be a finite idempotent algebra with pseudopartition operation* $x \vee y$*, such that every block of its semilattice replica congruence lies in the same tractable variety. Then* $\mathrm{CSP}(\mathbf{A})$ *is tractable.*

Proof.

Idea: Look in the biggest block possible! □

Theorem

$\mathcal{T}_2$ *has a pseudopartition operation* $(x \vee y = y(xy))$*, so it is tractable.*

Theorem

$\mathbb{T}_1$ *is the class of Płonka sums of squags, and it is tractable.*

Proof.

- Let $\Sigma = \{xx \approx x, xy \approx yx, x(x(yz)) \approx (x(xy))z\}$, and $x \vee y := y(xy)$.
- Squags satisfy $x \vee y \approx x$. Since $\mathbb{T}_1$ contains the variety of squags (i.e. squags satisfy $x(x(yz)) \approx (x(xy))z$), it is enough to show that Σ entails (P1)–(P5).
- Ask Prover9 to do it for you. Verify by hand over several days.

□

Theorem

$\mathbb{T}_1$ *is the class of Płonka sums of squags, and it is tractable.*

Proof.

- Let $\Sigma = \{xx \approx x, xy \approx yx, x(x(yz)) \approx (x(xy))z\}$, and $x \vee y := y(xy)$.
- Squags satisfy $x \vee y \approx x$. Since $\mathbb{T}_1$ contains the variety of squags (i.e. squags satisfy $x(x(yz)) \approx (x(xy))z$), it is enough to show that Σ entails (P1)–(P5).
- Ask Prover9 to do it for you. Verify by hand over several days.

□

Theorem

$\mathbb{T}_1$ is the class of Płonka sums of squags, and it is tractable.

Proof.

- Let $\Sigma = \{xx \approx x, xy \approx yx, x(x(yz)) \approx (x(xy))z\}$, and $x \vee y := y(xy)$.
- Squags satisfy $x \vee y \approx x$. Since $\mathbb{T}_1$ contains the variety of squags (i.e. squags satisfy $x(x(yz)) \approx (x(xy))z$), it is enough to show that Σ entails (P1)–(P5).
- Ask Prover9 to do it for you. Verify by hand over several days.

□

Theorem

$\mathbb{T}_1$ is the class of Płonka sums of squags, and it is tractable.

Proof.

- Let $\Sigma = \{xx \approx x, xy \approx yx, x(x(yz)) \approx (x(xy))z\}$, and $x \vee y := y(xy)$.
- Squags satisfy $x \vee y \approx x$. Since $\mathbb{T}_1$ contains the variety of squags (i.e. squags satisfy $x(x(yz)) \approx (x(xy))z$), it is enough to show that Σ entails (P1)–(P5).
- Ask Prover9 to do it for you. Verify by hand over several days.

□

Theorem

$\mathbb{T}_1$ is the class of Płonka sums of squags, and it is tractable.

Proof.

- Let $\Sigma = \{xx \approx x, xy \approx yx, x(x(yz)) \approx (x(xy))z\}$, and $x \vee y := y(xy)$.
- Squags satisfy $x \vee y \approx x$. Since $\mathbb{T}_1$ contains the variety of squags (i.e. squags satisfy $x(x(yz)) \approx (x(xy))z$), it is enough to show that Σ entails (P1)–(P5).
- Ask Prover9 to do it for you. Verify by hand over several days.

□

CID and CIE Groupoids

Definition

A groupoid is distributive (D) if it satisfies $x(yz) \approx (xy)(xz)$. It is entropic (E) if it satisfies $(xy)(zw) \approx (xz)(yw)$.

- Ježek, Kepka, and Němec: "the deepest non-associative theory within the framework of groupoids" is the theory of distributive groupoids.

Theorem

Every finite CID-groupoid (and hence CIE-groupoid) is a Płonka sum of quasigroups. The variety of CID groupoids is tractable.

CID and CIE Groupoids

Definition

A groupoid is distributive (D) if it satisfies $x(yz) \approx (xy)(xz)$. It is entropic (E) if it satisfies $(xy)(zw) \approx (xz)(yw)$.

- Ježek, Kepka, and Němec: "the deepest non-associative theory within the framework of groupoids" is the theory of distributive groupoids.

Theorem

Every finite CID-groupoid (and hence CIE-groupoid) is a Płonka sum of quasigroups. The variety of CID groupoids is tractable.

CID and CIE Groupoids

Definition

A groupoid is distributive (D) if it satisfies $x(yz) \approx (xy)(xz)$. It is entropic (E) if it satisfies $(xy)(zw) \approx (xz)(yw)$.

- Ježek, Kepka, and Němec: "the deepest non-associative theory within the framework of groupoids" is the theory of distributive groupoids.

Theorem

Every finite CID-groupoid (and hence CIE-groupoid) is a Płonka sum of quasigroups. The variety of CID groupoids is tractable.

Short Identities

Definition

A short groupoid identity $p \approx q$ is one in which

(i) the variables appearing in p and q are some subset of $\{x, y, z\}$

(ii) there are 3 or fewer variables appearing in p and q

(iii) no restriction is made to the ordering or grouping of the variables.

Theorem

There are four nontrivial varieties of CI-groupoids defined by an additional short identity: $\mathcal{S}q$ and $\mathcal{SL} \subseteq 2\mathcal{SL} \subseteq \mathcal{S}_3$, with $\mathcal{S}_3$ defined by

$$x(xy) \approx y(xy).$$

Theorem

$\mathcal{S}_3$ is congruence meet-semidistributive, and thus tractable.

Short Identities

Definition

A short groupoid identity $p \approx q$ is one in which

(i) the variables appearing in p and q are some subset of $\{x, y, z\}$

(ii) there are 3 or fewer variables appearing in p and q

(iii) no restriction is made to the ordering or grouping of the variables.

Theorem

There are four nontrivial varieties of CI-groupoids defined by an additional short identity: $\mathcal{S}q$ and $\mathcal{SL} \subseteq 2\mathcal{SL} \subseteq \mathcal{S}_3$, with $\mathcal{S}_3$ defined by

$$x(xy) \approx y(xy).$$

Theorem

$\mathcal{S}_3$ is congruence meet-semidistributive, and thus tractable.

Short Identities

Definition

A short groupoid identity $p \approx q$ is one in which

(i) the variables appearing in p and q are some subset of $\{x, y, z\}$

(ii) there are 3 or fewer variables appearing in p and q

(iii) no restriction is made to the ordering or grouping of the variables.

Theorem

There are four nontrivial varieties of CI-groupoids defined by an additional short identity: $\mathcal{S}q$ and $\mathcal{SL} \subseteq 2\mathcal{SL} \subseteq \mathcal{S}_3$, with $\mathcal{S}_3$ defined by

$$x(xy) \approx y(xy).$$

Theorem

$\mathcal{S}_3$ is congruence meet-semidistributive, and thus tractable.

Generalized Bol-Moufang Type Identities

Definition

An identity $p \approx q$ is of generalized Bol-Moufang type is one in which

(i) the same 3 variables appear in p and q,

(ii) one of the variables appears twice in p and q,

(iii) the remaining two variables appear once in p and q.

(The requirement that variables be ordered the same way in p and q is dropped.)

Theorem

Every variety of CI-groupoids of generalized Bol-Moufang type which is not of Bol-Moufang type is distributive, and thus tractable.

Further Research

- Other identities weaker than associativity?
- Finer structure of SD($\wedge$) varieties?
- CSP preservation results?