Algebraic Algorithms for the Inference Problem in Propositional Circumscription

Michał Wrona

Linköping University
SE-581 83 Linköping, Sweden
michal.wrona@liu.se

Algebraic and Model Theoretical Methods in Constraint Satisfaction, Banff 2014

Outline

Outline

Outline

Circumscription: Origins and Motivation

- Circumscription is a form of nonmonotonic reasoning introduced by McCarthy in *Circumscription — A Form of Nonmonotonic Reasoning. Artif. Intell. 1980*.
- In Google Scholar the paper has 2500 citations, the Schaefer's paper *The Complexity of Satisfiability Problems* — 1480.
- Circumscription was introduced in order to assure a common sense assumption that things are as expected unless otherwise specified
  - for instance by looking at minimal models.
- Used to solve frame problem in artificial intelligence.
- Closely related closed world reasoning, another nonmonotonic formalism.

Orders on Solutions to the CSP

$(D; \leq)$ a partial order; $\mathcal{I} = (V, D, C)$ an instance of the CSP with V part. into P, Z, Q.

Orders on Solutions to the CSP

$(D; \leq)$ a partial order; $\mathcal{I} = (V, D, C)$ an instance of the CSP with V part. into P, Z, Q.

Let $\alpha, \beta : V \to D$ solutions to $\mathcal{I}$.

- $\alpha \leq \beta$ if $\alpha(v) \leq \beta(v)$ for all $v \in V$;

Orders on Solutions to the CSP

$(D; \leq)$ a partial order; $\mathcal{I} = (V, D, C)$ an instance of the CSP with V part. into P, Z, Q.

Let $\alpha, \beta : V \to D$ solutions to $\mathcal{I}$.

- $\alpha \leq \beta$ if $\alpha(v) \leq \beta(v)$ for all $v \in V$;
- $\alpha \leq_{(P,Z)} \beta$ if
  - $\alpha(v) = \beta(v)$ for all $v \in Q$, and
  - $\alpha(v) \leq \beta(v)$ for all $v \in P$.

Orders on Solutions to the CSP

$(D; \leq)$ a partial order; $\mathcal{I} = (V, D, C)$ an instance of the CSP with V part. into P, Z, Q.

Let $\alpha, \beta : V \to D$ solutions to $\mathcal{I}$.

- $\alpha \leq \beta$ if $\alpha(v) \leq \beta(v)$ for all $v \in V$;
- $\alpha \leq_{(P,Z)} \beta$ if
  - $\alpha(v) = \beta(v)$ for all $v \in Q$, and
  - $\alpha(v) \leq \beta(v)$ for all $v \in P$.

Example

assignment	$p_1 \in P$	$p_2 \in P$	$z_1 \in Z$	$z_2 \in Z$	$q_1 \in Q$	$q_2 \in Q$
α	0	0	5	6	1	1
β	0	1	4	4	1	1
γ	0	0	7	7	0	0

We have: $\alpha <_{(P,Z)} \beta$, and γ incomparable to both α and β.

General Minimal Inference Problem

GMININF($\Gamma, (D; \leq)$)

$(D; \leq)$ a partial order and Γ a constraint language over D.

Instance: (V, P, Z, Q, D, C, ψ), where:

- $\mathcal{I} = (V, D, C)$ an instance of CSP(Γ) with V part. into P, Z, Q;
- ψ a constraint over var. in V.

Question: Is every $(\leq_{(P,Z)})$-minimal solution to $\mathcal{I}$ a solution to ψ?

General Minimal Inference Problem

GMININF($\Gamma, (D; \leq)$)

$(D; \leq)$ a partial order and Γ a constraint language over D.
Instance: (V, P, Z, Q, D, C, ψ), where:

- $\mathcal{I} = (V, D, C)$ an instance of CSP(Γ) with V part. into P, Z, Q;
- ψ a constraint over var. in V.

Question: Is every $(\leq_{(P,Z)})$-minimal solution to $\mathcal{I}$ a solution to ψ?

GMININF($\Gamma, (D; \leq)$) is always in $\Pi_2^P = \text{coNP}^{\text{NP}}$:

- We ask a counterexample: a $(\leq_{(P,Z)})$-min. sol. α to $\mathcal{I}$ which is not a sol. to ψ.
- We use the oracle to check if α is $(\leq_{(P,Z)})$-min.

General Minimal Inference Problem

GMININF$(\Gamma, (D; \leq))$

$(D; \leq)$ a partial order and Γ a constraint language over D.
Instance: (V, P, Z, Q, D, C, ψ), where:

- $\mathcal{I} = (V, D, C)$ an instance of CSP(Γ) with V part. into P, Z, Q;
- ψ a constraint over var. in V.

Question: Is every $(\leq_{(P,Z)})$-minimal solution to $\mathcal{I}$ a solution to ψ?

GMININF$(\Gamma, (D; \leq))$ is always in $\Pi_2^P = \text{coNP}^{\text{NP}}$:

- We ask a counterexample: a $(\leq_{(P,Z)})$-min. sol. α to $\mathcal{I}$ which is not a sol. to ψ.
- We use the oracle to check if α is $(\leq_{(P,Z)})$-min.

The complexity classified for $\{0, 1\}$ and $|_0^1$. Depend. on Γ: GMININF$(\Gamma, |_0^1)$ is Π_2^P-comp, coNP-comp, or in P. (many authors. the final step made by Durand, Hermann, Nordh, 2012.)

Versions of GMININF$(\Gamma, (D; \leq))$

- GMININF$(\Gamma, (D; \leq))$ with Z and Q are unrestricted,
- VMININF$(\Gamma, (D; \leq))$ with Z unrestricted and $Q = \emptyset$,
- CMININF$(\Gamma, (D; \leq))$ with $Z = \emptyset$ and Q unrestricted,
- MININF$(\Gamma, (D; \leq))$ with $Z = Q = \emptyset$.

Versions of GMININF($\Gamma, (D; \leq)$)

- GMININF($\Gamma, (D; \leq)$) with Z and Q are unrestricted,
- VMININF($\Gamma, (D; \leq)$) with Z unrestricted and $Q = \emptyset$,
- CMININF($\Gamma, (D; \leq)$) with $Z = \emptyset$ and Q unrestricted,
- MININF($\Gamma, (D; \leq)$) with $Z = Q = \emptyset$.

Complexity and polymorphisms. (Nordh, Jonsson, 2004.)

$(D; \leq)$ a partial order. Γ_1, Γ_2 over D such that Γ_2 has a pp-definition in Γ_1 (or equiv. $\mathrm{Pol}(\Gamma_1) \subseteq \mathrm{Pol}(\Gamma_2)$). Then we have that GMININF($\Gamma_2, (D; \leq)$) (VMININF($\Gamma_2, (D; \leq)$)) is polynomially time reducible to GMININF($\Gamma_1, (D; \leq)$) (VMININF($\Gamma_1, (D; \leq)$)).

Versions of $\mathrm{GMININF}(\Gamma, (D; \leq))$

- $\mathrm{GMININF}(\Gamma, (D; \leq))$ with Z and Q are unrestricted,
- $\mathrm{VMININF}(\Gamma, (D; \leq))$ with Z unrestricted and $Q = \emptyset$,
- $\mathrm{CMININF}(\Gamma, (D; \leq))$ with $Z = \emptyset$ and Q unrestricted,
- $\mathrm{MININF}(\Gamma, (D; \leq))$ with $Z = Q = \emptyset$.

Complexity and polymorphisms. (Nordh, Jonsson, 2004.)

$(D; \leq)$ a partial order. Γ_1, Γ_2 over D such that Γ_2 has a pp-definition in Γ_1 (or equiv. $\mathrm{Pol}(\Gamma_1) \subseteq \mathrm{Pol}(\Gamma_2)$). Then we have that $\mathrm{GMININF}(\Gamma_2, (D; \leq))$ ($\mathrm{VMININF}(\Gamma_2, (D; \leq))$) is polynomially time reducible to $\mathrm{GMININF}(\Gamma_1, (D; \leq))$ ($\mathrm{VMININF}(\Gamma_1, (D; \leq))$).

- For CMININF and MININF, one has to turn to partial clones.
- We concentrate on GMININF and VMININF.

Versions of GMININF($\Gamma, (D; \leq)$)

- GMININF($\Gamma, (D; \leq)$) with Z and Q are unrestricted,
- VMININF($\Gamma, (D; \leq)$) with Z unrestricted and $Q = \emptyset$,
- CMININF($\Gamma, (D; \leq)$) with $Z = \emptyset$ and Q unrestricted,
- MININF($\Gamma, (D; \leq)$) with $Z = Q = \emptyset$.

Complexity and polymorphisms. (Nordh, Jonsson, 2004.)

$(D; \leq)$ a partial order. Γ_1, Γ_2 over D such that Γ_2 has a pp-definition in Γ_1 (or equiv. $\mathrm{Pol}(\Gamma_1) \subseteq \mathrm{Pol}(\Gamma_2)$). Then we have that GMININF($\Gamma_2, (D; \leq)$) (VMININF($\Gamma_2, (D; \leq)$)) is polynomially time reducible to GMININF($\Gamma_1, (D; \leq)$) (VMININF($\Gamma_1, (D; \leq)$)).

- For CMININF and MININF, one has to turn to partial clones.
- We concentrate on GMININF and VMININF.

All the versions classified for $\{0, 1\}$.

GMININF over arbitrary finite domains

Nordh, Jonsson, *An Algebraic Approach to the Complexity of Propositional Circumscription*, LICS 2004.

- The authors consider GMININF(Γ), i.e., $(D; \leq)$ is a part of an input.

GMININF over arbitrary finite domains

Nordh, Jonsson, *An Algebraic Approach to the Complexity of Propositional Circumscription*, LICS 2004.

- The authors consider GMININF(Γ), i.e., $(D; \leq)$ is a part of an input.

What we can learn for our setting?

1. GMININF($\Gamma, (D; \leq)$) is polynomial time equivalent to GMININF($\Gamma \cup \{\{(d_1)\}, \ldots, \{(d_{|D|})\}\}, (D; \leq)$).

GMININF over arbitrary finite domains

Nordh, Jonsson, *An Algebraic Approach to the Complexity of Propositional Circumscription*, LICS 2004.

- The authors consider GMININF(Γ), i.e., $(D; \leq)$ is a part of an input.

What we can learn for our setting?

1. GMININF($\Gamma, (D; \leq)$) is polynomial time equivalent to GMININF($\Gamma \cup \{\{(d_1)\}, \ldots, \{(d_{|D|})\}\}, (D; \leq)$).
2. If Γ is conservative and CSP(Γ) is in P, then GMININF($\Gamma, (D; \leq)$) is in coNP for any $(D; \leq)$.

GMININF over arbitrary finite domains

Nordh, Jonsson, *An Algebraic Approach to the Complexity of Propositional Circumscription*, LICS 2004.

- The authors consider GMININF(Γ), i.e., $(D; \leq)$ is a part of an input.

What we can learn for our setting?

1. GMININF$(\Gamma, (D; \leq))$ is polynomial time equivalent to GMININF$(\Gamma \cup \{\{(d_1)\}, \ldots, \{(d_{|D|})\}\}, (D; \leq))$.
2. If Γ is conservative and CSP(Γ) is in P, then GMININF$(\Gamma, (D; \leq))$ is in coNP for any $(D; \leq)$.
3. If $(D; \leq)$ is linear and Alg(Γ) has a subalgebra with a two-element factor with projections only, then GMININF$(\Gamma, (D; \leq))$ is Π_2^P-complete.

GMININF over arbitrary finite domains

Nordh, Jonsson, *An Algebraic Approach to the Complexity of Propositional Circumscription*, LICS 2004.

- The authors consider GMININF(Γ), i.e., $(D; \leq)$ is a part of an input.

What we can learn for our setting?

1. GMININF$(\Gamma, (D; \leq))$ is polynomial time equivalent to GMININF$(\Gamma \cup \{\{(d_1)\}, \ldots, \{(d_{|D|})\}\}, (D; \leq))$.
2. If Γ is conservative and CSP(Γ) is in P, then GMININF$(\Gamma, (D; \leq))$ is in coNP for any $(D; \leq)$.
3. If $(D; \leq)$ is linear and Alg(Γ) has a subalgebra with a two-element factor with projections only, then GMININF$(\Gamma, (D; \leq))$ is Π_2^P-complete.

Corollary: If $(D; \leq)$ is linear and Γ is conservative, then GMININF$(\Gamma, (D; \leq))$ is Π_2^P-comp. or in coNP.

GMININF over arbitrary finite domains

Nordh, Jonsson, *An Algebraic Approach to the Complexity of Propositional Circumscription*, LICS 2004.

- The authors consider GMININF(Γ), i.e., $(D; \leq)$ is a part of an input.

What we can learn for our setting?

1. GMININF$(\Gamma, (D; \leq))$ is polynomial time equivalent to GMININF$(\Gamma \cup \{\{(d_1)\}, \ldots, \{(d_{|D|})\}\}, (D; \leq))$.
2. If Γ is conservative and CSP(Γ) is in P, then GMININF$(\Gamma, (D; \leq))$ is in coNP for any $(D; \leq)$.
3. If $(D; \leq)$ is linear and Alg(Γ) has a subalgebra with a two-element factor with projections only, then GMININF$(\Gamma, (D; \leq))$ is Π_2^P-complete.

Corollary: If $(D; \leq)$ is linear and Γ is conservative, then GMININF$(\Gamma, (D; \leq))$ is Π_2^P-comp. or in coNP.

Question:

- What about polynomial cases?

GMININF in Boolean Case

Theorem (Durand, Hermann, Nordh, 2012)

Γ over $\{0,1\}$. GMININF($\Gamma, |_0^1$) is in P if Γ is preserved by

1. both $\vee$ and $\wedge$ (clone M_2),
2. both $\wedge$ and $(x \wedge (\neg y \vee z))$ (clone S_{12}),
3. $((x \wedge z) \vee (x \wedge \neg y) \vee (z \wedge \neg y))$ (clone D_1).

Otherwise the problem GMININF($\Gamma, |_0^1$) is coNP-hard.

GMININF in Boolean Case

Theorem (Durand, Hermann, Nordh, 2012)

Γ over $\{0,1\}$. GMININF$(\Gamma, |_0^1)$ is in P if Γ is preserved by

1. both $\vee$ and $\wedge$ (clone M_2),
2. both $\wedge$ and $(x \wedge (\neg y \vee z))$ (clone S_{12}),
3. $((x \wedge z) \vee (x \wedge \neg y) \vee (z \wedge \neg y))$ (clone D_1).

Otherwise the problem GMININF$(\Gamma, |_0^1)$ is coNP-hard.

Our contributions. GMININF$(\Gamma, (D; \leq))$ is in P if

1. $(D; \leq)$ a lattice and Γ preserved by the join $\sqcup$ and the meet $\sqcap$.

GMININF in Boolean Case

Theorem (Durand, Hermann, Nordh, 2012)

Γ over $\{0,1\}$. GMININF$(\Gamma, |_0^1)$ is in P if Γ is preserved by

1. both $\vee$ and $\wedge$ (clone M_2),
2. both $\wedge$ and $(x \wedge (\neg y \vee z))$ (clone S_{12}),
3. $((x \wedge z) \vee (x \wedge \neg y) \vee (z \wedge \neg y))$ (clone D_1).

Otherwise the problem GMININF$(\Gamma, |_0^1)$ is coNP-hard.

Our contributions. GMININF$(\Gamma, (D; \leq))$ is in P if

1. $(D; \leq)$ a lattice and Γ preserved by the join $\sqcup$ and the meet $\sqcap$.
2. $(D; \leq)$ a meet-semilattice and Γ preserved by the meet $\sqcap$ and an S_{12}-operations, i.e., satisfying $f(y,x,x) = y$ and $f(y,y,x) = x$ for all $x \leq y$ in D.

GMININF in Boolean Case

Theorem (Durand, Hermann, Nordh, 2012)

Γ over $\{0,1\}$. GMININF$(\Gamma, |_0^1)$ is in P if Γ is preserved by

1. both $\vee$ and $\wedge$ (clone M_2),
2. both $\wedge$ and $(x \wedge (\neg y \vee z))$ (clone S_{12}),
3. $((x \wedge z) \vee (x \wedge \neg y) \vee (z \wedge \neg y))$ (clone D_1).

Otherwise the problem GMININF$(\Gamma, |_0^1)$ is coNP-hard.

Our contributions. GMININF$(\Gamma, (D; \leq))$ is in P if

1. $(D; \leq)$ a lattice and Γ preserved by the join $\sqcup$ and the meet $\sqcap$.
2. $(D; \leq)$ a meet-semilattice and Γ preserved by the meet $\sqcap$ and an S_{12}-operations, i.e., satisfying $f(y,x,x) = y$ and $f(y,y,x) = x$ for all $x \leq y$ in D.
3. $(D; \leq)$ is any order and Γ is preserved by a discriminator operation.

General Minimal Extension Problem

GMINEXT($\Gamma, (D; \leq)$)

$(D; \leq)$ a partial order and Γ a constraint language over D.
Instance: $(V, P, Z, Q, D, C, V_1, \alpha)$, where:

- $\mathcal{I} = (V, D, C)$ an instance of CSP(Γ) with V part. into P, Z, Q,
- $V_1 \subseteq V$ and $\alpha : V_1 \to D$.

Question: Is there a $(\leq_{(P,Z)})$-minimal solution to $\mathcal{I}$ extending α?

General Minimal Extension Problem

GMINEXT($\Gamma, (D; \leq)$)

$(D; \leq)$ a partial order and Γ a constraint language over D.
Instance: $(V, P, Z, Q, D, C, V_1, \alpha)$, where:

- $\mathcal{I} = (V, D, C)$ an instance of CSP(Γ) with V part. into P, Z, Q,
- $V_1 \subseteq V$ and $\alpha : V_1 \to D$.

Question: Is there a $(\leq_{(P,Z)})$-minimal solution to $\mathcal{I}$ extending α?

If GMINEXT($\Gamma, (D; \leq)$) in P, then GMININF($\Gamma, (D; \leq)$) in P. (Durand, Hermann, 2003.)

General Minimal Extension Problem

GMINEXT($\Gamma, (D; \leq)$)

$(D; \leq)$ a partial order and Γ a constraint language over D.
Instance: $(V, P, Z, Q, D, C, V_1, \alpha)$, where:

- $\mathcal{I} = (V, D, C)$ an instance of CSP(Γ) with V part. into P, Z, Q,
- $V_1 \subseteq V$ and $\alpha : V_1 \to D$.

Question: Is there a $(\leq_{(P,Z)})$-minimal solution to $\mathcal{I}$ extending α?

If GMINEXT($\Gamma, (D; \leq)$) in P, then GMININF($\Gamma, (D; \leq)$) in P. (Durand, Hermann, 2003.)

Algorithm for GMININF:

- Let (V, P, Z, Q, D, C, ψ) be an instance of GMININF($\Gamma, (D; \leq)$).
- If there is an assignment to Var(ψ) not satisfying ψ that can be extended to a $(\leq_{(P,Z)})$-minimal solution to $\mathcal{I}$, then return NO, else return YES.

Lattice, polymorphisms: $\sqcup$ and $\sqcap$.

Theorem

$(D; \leq)$ a lattice, Γ preserv. by $\sqcup$ and $\sqcap$, then $GMINEXT(\Gamma, (D; \leq))$ in P.

Lattice, polymorphisms: $\sqcup$ and $\sqcap$.

Theorem

$(D; \leq)$ a lattice, Γ preserv. by $\sqcup$ and $\sqcap$, then $GMINEXT(\Gamma, (D; \leq))$ in P.

Algorithm.

Instance: $(V, P, Z, Q, D, C, V_1, \alpha)$.

1. Let $Q_2 := Q \setminus V_1$.

Lattice, polymorphisms: $\sqcup$ and $\sqcap$.

Theorem

$(D; \leq)$ *a lattice,* Γ *preserv. by* $\sqcup$ *and* $\sqcap$*, then* $GMINEXT(\Gamma, (D; \leq))$ *in* P.

Algorithm.

Instance: $(V, P, Z, Q, D, C, V_1, \alpha)$.

1. Let $Q_2 := Q \setminus V_1$.
2. $\beta :=$ the greatest extension of α to $V_1 \cup Q_2$. (wrt. $(\leq)$)

Lattice, polymorphisms: $\sqcup$ and $\sqcap$.

Theorem

$(D; \leq)$ a lattice, Γ preserv. by $\sqcup$ and $\sqcap$, then $GMINEXT(\Gamma, (D; \leq))$ in P.

Algorithm.

Instance: $(V, P, Z, Q, D, C, V_1, \alpha)$.

1. Let $Q_2 := Q \setminus V_1$.
2. $\beta :=$ the greatest extension of α to $V_1 \cup Q_2$. (wrt. $(\leq)$)
3. If β does not exist return `false`.

Lattice, polymorphisms: $\sqcup$ and $\sqcap$.

Theorem

$(D; \leq)$ a lattice, Γ preserv. by $\sqcup$ and $\sqcap$, then $GMINEXT(\Gamma, (D; \leq))$ in P.

Algorithm.

Instance: $(V, P, Z, Q, D, C, V_1, \alpha)$.

1. Let $Q_2 := Q \setminus V_1$.
2. $\beta :=$ the greatest extension of α to $V_1 \cup Q_2$. (wrt. $(\leq)$)
3. If β does not exist return `false`.
4. $\gamma :=$ the least extension of β to V.

Lattice, polymorphisms: $\sqcup$ and $\sqcap$.

Theorem

$(D; \leq)$ a lattice, Γ preserv. by $\sqcup$ and $\sqcap$, then $GMINEXT(\Gamma, (D; \leq))$ in P.

Algorithm.

Instance: $(V, P, Z, Q, D, C, V_1, \alpha)$.

1. Let $Q_2 := Q \setminus V_1$.
2. $\beta :=$ the greatest extension of α to $V_1 \cup Q_2$. (wrt. $(\leq)$)
3. If β does not exist return `false`.
4. $\gamma :=$ the least extension of β to V.
5. If γ does not exist return `false`.

Lattice, polymorphisms: $\sqcup$ and $\sqcap$.

Theorem

$(D; \leq)$ a lattice, Γ preserv. by $\sqcup$ and $\sqcap$, then $GMINEXT(\Gamma, (D; \leq))$ in P.

Algorithm.

Instance: $(V, P, Z, Q, D, C, V_1, \alpha)$.

1. Let $Q_2 := Q \setminus V_1$.
2. $\beta :=$ the greatest extension of α to $V_1 \cup Q_2$. (wrt. $(\leq)$)
3. If β does not exist return `false`.
4. $\gamma :=$ the least extension of β to V.
5. If γ does not exist return `false`.
6. $\theta :=$ the least extension of $\gamma|_Q$ to V.

Lattice, polymorphisms: $\sqcup$ and $\sqcap$.

Theorem

$(D; \leq)$ a lattice, Γ preserv. by $\sqcup$ and $\sqcap$, then GMINEXT$(\Gamma, (D; \leq))$ in P.

Algorithm.

Instance: $(V, P, Z, Q, D, C, V_1, \alpha)$.

1. Let $Q_2 := Q \setminus V_1$.
2. $\beta :=$ the greatest extension of α to $V_1 \cup Q_2$. (wrt. $(\leq)$)
3. If β does not exist return `false`.
4. $\gamma :=$ the least extension of β to V.
5. If γ does not exist return `false`.
6. $\theta :=$ the least extension of $\gamma|_Q$ to V.
7. If $\gamma|_P = \theta|_P$ return `true`
8. else return `false`.

Lattice, polymorphisms: $\sqcup$ and $\sqcap$.

Theorem

$(D; \leq)$ a lattice, Γ preserv. by $\sqcup$ and $\sqcap$, then GMINEXT$(\Gamma, (D; \leq))$ in P.

Algorithm.

Instance: $(V, P, Z, Q, D, C, V_1, \alpha)$.

1. Let $Q_2 := Q \setminus V_1$.
2. $\beta :=$ the greatest extension of α to $V_1 \cup Q_2$. (wrt. $(\leq)$)
3. If β does not exist return `false`.
4. $\gamma :=$ the least extension of β to V.
5. If γ does not exist return `false`.
6. $\theta :=$ the least extension of $\gamma|_Q$ to V.
7. If $\gamma|_P = \theta|_P$ return `true`
8. else return `false`.

$(D; \leq)$ lattice. Relation preserved by both $\sqcap$ and $\sqcup$ iff definable by a conjunction of clauses of the form: $(x \leq d_1 \vee y \geq d_2)$.

Meet-semilattice, polymorphisms: $\sqcap$, S_{12}-operation

Theorem

$(D; \leq)$ a meet-semilattice, Γ preserv. by $\sqcap$ and an S_{12}-operation, then $GMINEXT(\Gamma, (D; \leq))$ in P.

Meet-semilattice, polymorphisms: $\sqcap$, S_{12}-operation

Theorem

$(D; \leq)$ a meet-semilattice, Γ preserv. by $\sqcap$ and an S_{12}-operation, then $GMINEXT(\Gamma, (D; \leq))$ in P.

Algorithm.

Instance: $(V, P, Z, Q, D, C, V_1, \alpha)$.

Meet-semilattice, polymorphisms: $\sqcap$, S_{12}-operation

Theorem

$(D; \leq)$ a meet-semilattice, Γ preserv. by $\sqcap$ and an S_{12}-operation, then $GMINEXT(\Gamma, (D; \leq))$ in P.

Algorithm.

Instance: $(V, P, Z, Q, D, C, V_1, \alpha)$.

1. $\gamma :=$ the least extension of α to V.
2. If γ does not exist return `false`.

Meet-semilattice, polymorphisms: $\sqcap$, S_{12}-operation

Theorem

$(D; \leq)$ a meet-semilattice, Γ preserv. by $\sqcap$ and an S_{12}-operation, then $GMINEXT(\Gamma, (D; \leq))$ in P.

Algorithm.

Instance: $(V, P, Z, Q, D, C, V_1, \alpha)$.

1. $\gamma :=$ the least extension of α to V.
2. If γ does not exist return `false`.
3. $\theta :=$ the least extension of $\gamma|_Q$ to V.
4. If $\gamma|_P = \theta|_P$ return `true`
5. else return `false`.

Meet-semilattice, polymorphisms: $\sqcap$, S_{12}-operation

Theorem

$(D; \leq)$ a meet-semilattice, Γ preserv. by $\sqcap$ and an S_{12}-operation, then $GMINEXT(\Gamma, (D; \leq))$ in P.

Algorithm.

Instance: $(V, P, Z, Q, D, C, V_1, \alpha)$.

1. $\gamma :=$ the least extension of α to V.
2. If γ does not exist return `false`.
3. $\theta :=$ the least extension of $\gamma|_Q$ to V.
4. If $\gamma|_P = \theta|_P$ return `true`
5. else return `false`.

We use an S_{12}-operation in the proof.

Meet-semilattice, polymorphisms: $\sqcap$, S_{12}-operation

Theorem

$(D; \leq)$ a meet-semilattice, Γ preserv. by $\sqcap$ and an S_{12}-operation, then $GMINEXT(\Gamma, (D; \leq))$ in P.

Algorithm.

Instance: $(V, P, Z, Q, D, C, V_1, \alpha)$.

1. $\gamma :=$ the least extension of α to V.
2. If γ does not exist return `false`.
3. $\theta :=$ the least extension of $\gamma|_Q$ to V.
4. If $\gamma|_P = \theta|_P$ return `true`
5. else return `false`.

We use an S_{12}-operation in the proof.

Some relations preserv. by a S_{12}-op and $\sqcap$

$(D; \leq)$ a lattice. Every relation definable by a conjunction of clauses of the form $(x \geq d)$ and $(x_1 \leq d_1 \vee \cdots \vee x_k \leq d_k)$ is preserved by a S_{12}-operation and $\sqcap$.

VMININF in Boolean Case

Theorem (Durand, Hermann, Nordh, 2012)

Γ over $\{0, 1\}$. VMININF($\Gamma, |^1_0$) is in P if Γ is preserved by

1. $\wedge$,
2. constant operation 0 and one of: $\vee$, minority or majority,
3. $((x \wedge z) \vee (x \wedge \neg y) \vee (z \wedge \neg y))$.

Otherwise, the problem VMININF($\Gamma, |^1_0$) is coNP-hard.

VMININF in Boolean Case

Theorem (Durand, Hermann, Nordh, 2012)

Γ over $\{0, 1\}$. VMININF$(\Gamma, |_0^1)$ is in P if Γ is preserved by

1. $\wedge$,
2. constant operation 0 and one of: $\vee$, minority or majority,
3. $((x \wedge z) \vee (x \wedge \neg y) \vee (z \wedge \neg y))$.

Otherwise, the problem VMININF$(\Gamma, |_0^1)$ is coNP-hard.

Our contributions. VMININF$(\Gamma, (D; \leq))$ is in P if

VMININF in Boolean Case

Theorem (Durand, Hermann, Nordh, 2012)

Γ over $\{0,1\}$. VMININF$(\Gamma, |_0^1)$ is in P if Γ is preserved by

1. $\wedge$,
2. constant operation 0 and one of: $\vee$, minority or majority,
3. $((x \wedge z) \vee (x \wedge \neg y) \vee (z \wedge \neg y))$.

Otherwise, the problem VMININF$(\Gamma, |_0^1)$ is coNP-hard.

Our contributions. VMININF$(\Gamma, (D; \leq))$ is in P if

1. $(D; \leq)$ a meet-semilattice and Γ preserved by the meet $\sqcap$.

VMININF in Boolean Case

Theorem (Durand, Hermann, Nordh, 2012)

Γ over $\{0,1\}$. VMININF($\Gamma, |^1_0$) is in P if Γ is preserved by

1. $\wedge$,
2. constant operation 0 and one of: $\vee$, minority or majority,
3. $((x \wedge z) \vee (x \wedge \neg y) \vee (z \wedge \neg y))$.

Otherwise, the problem VMININF($\Gamma, |^1_0$) is coNP-hard.

Our contributions. VMININF($\Gamma, (D; \leq)$) is in P if

1. $(D; \leq)$ a meet-semilattice and Γ preserved by the meet $\sqcap$.
2. $(D; \leq)$ with the least element $\perp$ and Γ preserved by a constant operation $\perp$ and an idempotent operation f such that CSP(Γ) is in P whenever Γ preserved by f.

VMININF in Boolean Case

Theorem (Durand, Hermann, Nordh, 2012)

Γ over $\{0,1\}$. VMININF($\Gamma, |_0^1$) is in P if Γ is preserved by

1. $\wedge$,
2. constant operation 0 and one of: $\vee$, minority or majority,
3. $((x \wedge z) \vee (x \wedge \neg y) \vee (z \wedge \neg y))$.

Otherwise, the problem VMININF($\Gamma, |_0^1$) is coNP-hard.

Our contributions. VMININF($\Gamma, (D; \leq)$) is in P if

1. $(D; \leq)$ a meet-semilattice and Γ preserved by the meet $\sqcap$.
2. $(D; \leq)$ with the least element $\perp$ and Γ preserved by a constant operation $\perp$ and an idempotent operation f such that CSP(Γ) is in P whenever Γ preserved by f.
3. $(D; \leq)$ is any order and Γ is preserved by a discriminator operation.

Conditions for coNP Containment

Theorem (Durand, Hermann, Nordh, 2012)

Γ over $\{0,1\}$. GMININF(Γ, $|^1_0$) is in coNP if Γ is preserved by one of the following: $\wedge$, $\vee$, majority, or minority. Otherwise the problem GMININF(Γ, $|^1_0$) is Π_2^P-hard.

Conditions for coNP Containment

Theorem (Durand, Hermann, Nordh, 2012)

Γ over $\{0,1\}$. GMININF($\Gamma, |_0^1$) is in coNP if Γ is preserved by one of the following: $\wedge$, $\vee$, majority, or minority. Otherwise the problem GMININF($\Gamma, |_0^1$) is Π_2^P-hard.

Our contributions. GMININF($\Gamma, (D; \leq)$) is in coNP if

1. $(D; \leq)$ a meet-semilattice and Γ preserved by the meet $\sqcap$.
2. $(D; \leq)$ a join-semilattice and Γ preserved by the join $\sqcup$.

Conditions for coNP Containment

Theorem (Durand, Hermann, Nordh, 2012)

Γ over $\{0, 1\}$. GMININF($\Gamma, |^1_0$) is in coNP if Γ is preserved by one of the following: $\wedge$, $\vee$, majority, or minority. Otherwise the problem GMININF($\Gamma, |^1_0$) is Π_2^P-hard.

Our contributions. GMININF($\Gamma, (D; \leq)$) is in coNP if

1. $(D; \leq)$ a meet-semilattice and Γ preserved by the meet $\sqcap$.
2. $(D; \leq)$ a join-semilattice and Γ preserved by the join $\sqcup$.

But we do not know what happens for near-unanimity and Maltsev operations.

Conditions for coNP Containment

Theorem (Durand, Hermann, Nordh, 2012)

Γ over $\{0,1\}$. GMININF($\Gamma, |_0^1$) is in coNP if Γ is preserved by one of the following: $\wedge$, $\vee$, majority, or minority. Otherwise the problem GMININF($\Gamma, |_0^1$) is Π_2^P-hard.

Our contributions. GMININF($\Gamma, (D; \leq)$) is in coNP if

1. $(D; \leq)$ a meet-semilattice and Γ preserved by the meet $\sqcap$.
2. $(D; \leq)$ a join-semilattice and Γ preserved by the join $\sqcup$.

But we do not know what happens for near-unanimity and Maltsev operations.

VMININF($\Gamma, |_0^1$) is also in coNP if Γ is preserved by the constant operation 0.

Conditions for coNP Containment

Theorem (Durand, Hermann, Nordh, 2012)

Γ over $\{0,1\}$. GMININF($\Gamma, |_0^1$) is in coNP if Γ is preserved by one of the following: $\wedge$, $\vee$, majority, or minority. Otherwise the problem GMININF($\Gamma, |_0^1$) is Π_2^P-hard.

Our contributions. GMININF($\Gamma, (D; \leq)$) is in coNP if

1. $(D; \leq)$ a meet-semilattice and Γ preserved by the meet $\sqcap$.
2. $(D; \leq)$ a join-semilattice and Γ preserved by the join $\sqcup$.

But we do not know what happens for near-unanimity and Maltsev operations.

VMININF($\Gamma, |_0^1$) is also in coNP if Γ is preserved by the constant operation 0.

We can prove: VMININF($\Gamma, (D; \leq)$) is in coNP if $(D; \leq)$ has the least element $\perp$ and Γ is preserved by the constant operation $\perp$.

Future Work

- Further generalizations of tractable cases.

Future Work

- Further generalizations of tractable cases.
- Conditions for coNP containment.

Future Work

- Further generalizations of tractable cases.
- Conditions for coNP containment.
- Conditions for hardness.

Future Work

- Further generalizations of tractable cases.
- Conditions for coNP containment.
- Conditions for hardness.
- Partial classifications: conservative case, three element case.

Future Work

- Further generalizations of tractable cases.
- Conditions for coNP containment.
- Conditions for hardness.
- Partial classifications: conservative case, three element case.

Thank you for your attention.