Relating Semantic and Proof-Theoretic Concepts for Polynomial Time Decidability of Uniform Word Problems

Harald Ganzinger

Max-Planck-Institut für Informatik

Introduction

- PTIME decidable uniform word problems for quasi-varieties
- CS applications: type inference systems, program analysis, decision procedures in ATP
- fundamental effective methods for establishing PTIME upper bounds
- local theories (Givan, McAllester 92) capture PTIME
- algebraic criteria by Skolem (1920), Evans (1951), and Burris (1995)
- relation between these criteria

Introduction

- PTIME decidable uniform word problems for quasi-varieties
- CS applications: type inference systems, program analysis, decision procedures in ATP
- fundamental effective methods for establishing PTIME upper bounds
- local theories (Givan, McAllester 92) capture PTIME
- algebraic criteria by Skolem (1920), Evans (1951), and Burris (1995)
- relation between these criteria

Introduction

- PTIME decidable uniform word problems for quasi-varieties
- CS applications: type inference systems, program analysis, decision procedures in ATP
- fundamental effective methods for establishing PTIME upper bounds
- local theories (Givan, McAllester 92) capture PTIME
- algebraic criteria by Skolem (1920), Evans (1951), and Burris (1995)
- relation between these criteria

Introduction

- PTIME decidable uniform word problems for quasi-varieties
- CS applications: type inference systems, program analysis, decision procedures in ATP
- fundamental effective methods for establishing PTIME upper bounds
- local theories (Givan, McAllester 92) capture PTIME
- algebraic criteria by Skolem (1920), Evans (1951), and Burris (1995)
- relation between these criteria

Introduction

- PTIME decidable uniform word problems for quasi-varieties
- CS applications: type inference systems, program analysis, decision procedures in ATP
- fundamental effective methods for establishing PTIME upper bounds
- local theories (Givan, McAllester 92) capture PTIME
- algebraic criteria by Skolem (1920), Evans (1951), and Burris (1995)
- relation between these criteria

Introduction

- PTIME decidable uniform word problems for quasi-varieties
- CS applications: type inference systems, program analysis, decision procedures in ATP
- fundamental effective methods for establishing PTIME upper bounds
- local theories (Givan, McAllester 92) capture PTIME
- algebraic criteria by Skolem (1920), Evans (1951), and Burris (1995)
- relation between these criteria

Example: Theories of Data Structures

Natural numbers with inequality

$$
\begin{array}{rcl}
s(n) \doteq s(m) & \rightarrow & n \doteq m \\
 & \rightarrow & (0 < s(n)) \doteq \top \\
(s(n) < 0) \doteq \top & \rightarrow & \\
(n < m) \doteq \top & \rightarrow & (s(n) < s(m)) \doteq \top \\
(s(n) < s(m)) \doteq \top & \rightarrow & (n < m) \doteq \top
\end{array}
$$

Lists

$$
\begin{array}{cl}
\rightarrow & \mathsf{car}(\mathsf{cons}(x, y)) \doteq x \\
\rightarrow & \mathsf{cdr}(\mathsf{cons}(x, y)) \doteq y \\
\rightarrow & \mathsf{length}(\mathsf{nil}) \doteq 0 \\
\rightarrow & \mathsf{length}(\mathsf{cons}(x, y)) \doteq s(\mathsf{length}(y))
\end{array}
$$

Terminology

logic	algebra
universal Horn theory $\mathcal{K}$	quasi-variety $\mathcal{K}$−alg
entailment problem $\mathcal{K} \models C$ for universal/ground Horn clauses C	uniform word problem
query $C = \Gamma \rightarrow s \doteq t$	defining relations Γ, and word problem $s \doteq t$
var(C) (= Skolem constants of $\neg C$)	generators

Skolem and Datalog (1920)

Given $\mathcal{K} \overset{?}{\models} C$, consider the relational approximation $\mathcal{K}^* \overset{?}{\models} C^*$:

Skolem and Datalog (1920)

Given $\mathcal{K} \overset{?}{\models} C$, consider the relational approximation $\mathcal{K}^* \overset{?}{\models} C^*$:

- flatten clauses by term abstraction $C[t] \;\Rightarrow\; x \dot{\neq} t \vee C[x]$

Skolem and Datalog (1920)

Given $\mathcal{K} \overset{?}{\models} C$, consider the relational approximation $\mathcal{K}^* \overset{?}{\models} C^*$:

- flatten clauses by term abstraction $C[t] \ \Rightarrow \ x \not\doteq t \vee C[x]$
- replace functions f by relations r_f

Skolem and Datalog (1920)

Given $\mathcal{K} \overset{?}{\models} C$, consider the relational approximation $\mathcal{K}^* \overset{?}{\models} C^*$:

- flatten clauses by term abstraction $C[t] \;\Rightarrow\; x \not\doteq t \vee C[x]$
- replace functions f by relations r_f
- replace equations $y \doteq f(\vec{x})$ by atoms $r_f(\vec{x}, y)$

Skolem and Datalog (1920)

Given $\mathcal{K} \stackrel{?}{\models} C$, consider the relational approximation $\mathcal{K}^* \stackrel{?}{\models} C^*$:

- flatten clauses by term abstraction $C[t] \;\Rightarrow\; x \not\doteq t \vee C[x]$
- replace functions f by relations r_f
- replace equations $y \doteq f(\vec{x})$ by atoms $r_f(\vec{x}, y)$
- require relations to be unambiguous by adding clauses
  $r_f(\vec{x}, y), r_f(\vec{x}, z) \rightarrow y \doteq z$

Skolem and Datalog (1920)

Given $\mathcal{K} \overset{?}{\models} C$, consider the relational approximation $\mathcal{K}^* \overset{?}{\models} C^*$:

- flatten clauses by term abstraction $C[t] \;\Rightarrow\; x \not\doteq t \vee C[x]$
- replace functions f by relations r_f
- replace equations $y \doteq f(\vec{x})$ by atoms $r_f(\vec{x}, y)$
- require relations to be unambiguous by adding clauses
  $r_f(\vec{x}, y), r_f(\vec{x}, z) \rightarrow y \doteq z$
- add the congruence axioms for $\doteq$

Skolem and Datalog (1920)

Given $\mathcal{K} \overset{?}{\models} C$, consider the relational approximation $\mathcal{K}^* \overset{?}{\models} C^*$:

- flatten clauses by term abstraction $C[t] \;\Rightarrow\; x \not\doteq t \vee C[x]$
- replace functions f by relations r_f
- replace equations $y \doteq f(\vec{x})$ by atoms $r_f(\vec{x}, y)$
- require relations to be unambiguous by adding clauses $r_f(\vec{x}, y), r_f(\vec{x}, z) \rightarrow y \doteq z$
- add the congruence axioms for $\doteq$
- soundness: $\mathcal{K}^* \models C^*$ implies $\mathcal{K} \models C$

Skolem and Datalog (1920)

Given $\mathcal{K} \stackrel{?}{\models} C$, consider the relational approximation $\mathcal{K}^* \stackrel{?}{\models} C^*$:

- flatten clauses by term abstraction $C[t] \;\Rightarrow\; x \not\doteq t \vee C[x]$
- replace functions f by relations r_f
- replace equations $y \doteq f(\vec{x})$ by atoms $r_f(\vec{x}, y)$
- require relations to be unambiguous by adding clauses $r_f(\vec{x}, y), r_f(\vec{x}, z) \rightarrow y \doteq z$
- add the congruence axioms for $\doteq$
- soundness: $\mathcal{K}^* \models C^*$ implies $\mathcal{K} \models C$
- completeness (I): $\mathcal{K}^* \not\models C^*$ implies the existence of a finite $\mathcal{K}^*$-structure in which C^* is false

Skolem and Datalog (1920)

Given $\mathcal{K} \overset{?}{\models} C$, consider the relational approximation $\mathcal{K}^* \overset{?}{\models} C^*$:

- flatten clauses by term abstraction $C[t] \;\Rightarrow\; x \not\doteq t \vee C[x]$
- replace functions f by relations r_f
- replace equations $y \doteq f(\vec{x})$ by atoms $r_f(\vec{x}, y)$
- require relations to be unambiguous by adding clauses $r_f(\vec{x}, y), r_f(\vec{x}, z) \rightarrow y \doteq z$
- add the congruence axioms for $\doteq$
- soundness: $\mathcal{K}^* \models C^*$ implies $\mathcal{K} \models C$
- completeness (I): $\mathcal{K}^* \not\models C^*$ implies the existence of a finite $\mathcal{K}^*$-structure in which C^* is false
- completeness (II): if every finite $\mathcal{K}^*$-structure can be embedded into a $\mathcal{K}^*$-structure in which all relations are total, then $\mathcal{K}^* \not\models C^*$ implies $\mathcal{K} \not\models C$

Evans' Embeddability Criterion

THEOREM [Evans 51, Burris 95] Let $\mathcal{K}$ be a finite set of Horn clauses. If every finite partial $\mathcal{K}$-algebra weakly embeds into $\mathcal{K}$, then the uniform word problem for $\mathcal{K}$ is decidable in polynomial time.

Examples [Skolem 1920]: lattices, fragments of geometry

Note: embeddability is a property of a presentation, not of a quasi-variety

Partial Algebras from Relational Structures

- functions may be partial

Partial Algebras from Relational Structures

- functions may be partial
- weak homomorphisms $h : A \longrightarrow B$:
  if $f_A(\vec{a})$ defined then $f_B(h(\vec{a}))$ defined (but not vice-versa)

Partial Algebras from Relational Structures

- functions may be partial
- weak homomorphisms $h : A \to B$:

  if $f_A(\vec{a})$ defined then $f_B(h(\vec{a}))$ defined (but not vice-versa)
- Evans' definition of (strong) truth:

$$A, \beta \models s_1 \doteq t_1, \ldots, s_k \doteq t_k \to s \doteq t,$$

if whenever the $\beta(s_i)$ and $\beta(t_i)$ are defined and equal, then

(i) if $\beta(s)$ and $\beta(t)$ are defined then they are equal

(ii) if $s = f(u_1, \ldots, u_n)$, and if the $\beta(u_i)$ and $\beta(t)$ are defined, then $\beta(s)$ is also defined.

Partial Algebras from Relational Structures

- functions may be partial
- weak homomorphisms $h : A \to B$:
  if $f_A(\vec{a})$ defined then $f_B(h(\vec{a}))$ defined (but not vice-versa)
- Evans' definition of (strong) truth:

$$A, \beta \models s_1 \doteq t_1, \ldots, s_k \doteq t_k \to s \doteq t,$$

  if whenever the $\beta(s_i)$ and $\beta(t_i)$ are defined and equal, then
  (i) if $\beta(s)$ and $\beta(t)$ are defined then they are equal
  (ii) if $s = f(u_1, \ldots, u_n)$, and if the $\beta(u_i)$ and $\beta(t)$ are defined, then $\beta(s)$ is also defined.
- in the relational encoding (i) is automatic, (ii) can be enforced

Example

Ad (i)

$$C = \; y \doteq a \;\rightarrow\; f(x,y) \doteq g(f(y,x),y)$$

becomes

$$C^* = \; r_a(z), y \doteq z, r_f(x,y,xy), r_f(y,x,yx), r_g(yx,y,u) \;\rightarrow\; xy \doteq u$$

Therefore, $A \models_{(\text{i})} C$ iff $A^* \models C^*$.

Example

Ad (i)

$$C = \ y \doteq a \ \rightarrow \ f(x, y) \doteq g(f(y, x), y)$$

becomes

$$C^* = \ r_a(z), y \doteq z, r_f(x, y, xy), r_f(y, x, yx), r_g(yx, y, u) \ \rightarrow \ xy \doteq u$$

Therefore, $A \models_{(\mathrm{i})} C$ iff $A^* \models C^*$.

Ad (ii) this can be expressed in Datalog by adding

$r_a(z), y \doteq z, r_f(x, y, xy), r_f(y, x, yx) \rightarrow r_g(yx, y, xy)$

$r_a(z), y \doteq z, r_f(y, x, yx), r_g(yx, y, u) \rightarrow r_f(x, y, u)$

Example

Ad (i)

$$C = \; y \doteq a \;\rightarrow\; f(x,y) \doteq g(f(y,x),y)$$

becomes

$$C^* = \; r_a(z), y \doteq z, r_f(x,y,xy), r_f(y,x,yx), r_g(yx,y,u) \;\rightarrow\; xy \doteq u$$

Therefore, $A \models_{(\mathrm{i})} C$ iff $A^* \models C^*$.

Ad (ii) this can be expressed in Datalog by adding

$r_a(z), y \doteq z, r_f(x,y,xy), r_f(y,x,yx) \rightarrow r_g(yx,y,xy)$
$r_a(z), y \doteq z, r_f(y,x,yx), r_g(yx,y,u) \rightarrow r_f(x,y,u)$

Conclusion: Evans' "strong truth" is the canonical concept of truth from the Datalog point of view.

Local Theories [Givan, McAllester 92]

Definition: $\mathcal{K}$ is local, if for ground Horn clauses C

$$\mathcal{K} \models C \quad \text{iff} \quad \mathcal{K}_{\mathsf{st}(C)} \models C,$$

where $\mathcal{K}_{\mathsf{st}(C)}$ the set of ground instances of $\mathcal{K}$ were each term is a subterm of C.

Local Theories [Givan, McAllester 92]

Definition: $\mathcal{K}$ is local, if for ground Horn clauses C

$$\mathcal{K} \models C \quad \text{iff} \quad \mathcal{K}_{\mathsf{st}(C)} \models C,$$

where $\mathcal{K}_{\mathsf{st}(C)}$ the set of ground instances of $\mathcal{K}$ were each term is a subterm of C.

Results: - locality captures PTIME

- co-recursively enumerable but undecidable
- related to saturation by ordered resolution

Local Theories [Givan, McAllester 92]

Definition: $\mathcal{K}$ is local, if for ground Horn clauses C

$$\mathcal{K} \models C \quad \text{iff} \quad \mathcal{K}_{\mathsf{st}(C)} \models C,$$

where $\mathcal{K}_{\mathsf{st}(C)}$ the set of ground instances of $\mathcal{K}$ were each term is a subterm of C.

Results: - locality captures PTIME

- co-recursively enumerable but undecidable
- related to saturation by ordered resolution

Examples: congruence closure, lattices, atomic set constraints, embeddability of terms, joinability in ground rewrite systems

Local Theories [Givan, McAllester 92]

Definition: $\mathcal{K}$ is local, if for ground Horn clauses C

$$\mathcal{K} \models C \quad \text{iff} \quad \mathcal{K}_{\mathsf{st}(C)} \models C,$$

where $\mathcal{K}_{\mathsf{st}(C)}$ the set of ground instances of $\mathcal{K}$ were each term is a subterm of C.

Results: - locality captures PTIME
- co-recursively enumerable but undecidable
- related to saturation by ordered resolution

Examples: congruence closure, lattices, atomic set constraints, embeddability of terms, joinability in ground rewrite systems

Previously: only non-equational case considered

Specifics of Local Equational Theories

Definition: same (work with equational logic)

Specifics of Local Equational Theories

Definition: same (work with equational logic)

Results: basically the same

Specifics of Local Equational Theories

Definition: same (work with equational logic)

Results: basically the same

Example:

$$
\begin{aligned}
p(x) \doteq y &\quad\longrightarrow\quad s(y) \doteq x \\
s(x) \doteq y &\quad\longrightarrow\quad p(y) \doteq x \\
p(x) \doteq p(y) &\quad\longrightarrow\quad x \doteq y \\
s(x) \doteq s(y) &\quad\longrightarrow\quad x \doteq y
\end{aligned}
$$

Specifics of Local Equational Theories

Definition: same (work with equational logic)

Results: basically the same

Example:

$$
\begin{aligned}
p(x) \doteq y &\quad\longrightarrow\quad s(y) \doteq x \\
s(x) \doteq y &\quad\longrightarrow\quad p(y) \doteq x \\
p(x) \doteq p(y) &\quad\longrightarrow\quad x \doteq y \\
s(x) \doteq s(y) &\quad\longrightarrow\quad x \doteq y
\end{aligned}
$$

Counterexample:

$$
s(p(x)) \doteq x \qquad\qquad p(s(x)) \doteq x
$$

Specifics of Local Equational Theories

Definition: same (work with equational logic)

Results: basically the same

Example:

$$
\begin{aligned}
p(x) \doteq y \quad &\longrightarrow \quad s(y) \doteq x \\
s(x) \doteq y \quad &\longrightarrow \quad p(y) \doteq x \\
p(x) \doteq p(y) \quad &\longrightarrow \quad x \doteq y \\
s(x) \doteq s(y) \quad &\longrightarrow \quad x \doteq y
\end{aligned}
$$

Counterexample:

$$
s(p(x)) \doteq x \qquad\qquad p(s(x)) \doteq x
$$

Observation: irredundant local theories have flat clauses only

Specifics of Local Equational Theories (II)

Theorem $\mathcal{K}$ is local in equational logic
iff $\mathcal{K} \cup EQ$ is local in non-equational logic.

Specifics of Local Equational Theories (II)

THEOREM $\mathcal{K}$ is local in equational logic
iff $\mathcal{K} \cup EQ$ is local in non-equational logic.

So what? better complexity if equality is internal

Specifics of Local Equational Theories (II)

THEOREM $\mathcal{K}$ is local in equational logic
iff $\mathcal{K} \cup EQ$ is local in non-equational logic.

So what? better complexity if equality is internal

Stable locality: allow theory variables to be instantiated by arbitrary query subterms

Specifics of Local Equational Theories (II)

THEOREM $\mathcal{K}$ is local in equational logic
iff $\mathcal{K} \cup EQ$ is local in non-equational logic.

So what? better complexity if equality is internal

Stable locality: allow theory variables to be instantiated by arbitrary query subterms

Example:

$$s(p(x)) \doteq x \qquad p(s(x)) \doteq x$$

Specifics of Local Equational Theories (II)

THEOREM $\mathcal{K}$ is local in equational logic
iff $\mathcal{K} \cup EQ$ is local in non-equational logic.

So what? better complexity if equality is internal

Stable locality: allow theory variables to be instantiated by arbitrary query subterms

Example:

$$s(p(x)) \doteq x \qquad p(s(x)) \doteq x$$

Locality vs stable locality: $\mathcal{K}$ local $\Rightarrow$ $\mathcal{K}$ stably local $\Rightarrow$ $\mathcal{K}'$ local
($\mathcal{K}'$ obtained from massaging $\mathcal{K}$)

Embeddability vs Locality

THEOREM $\mathcal{K}$ is local $\Rightarrow$ $\mathcal{K}$ weakly embeds $\Rightarrow$ $\mathcal{K}$ is stably local

Embeddability vs Locality

THEOREM $\mathcal{K}$ is local $\Rightarrow$ $\mathcal{K}$ weakly embeds $\Rightarrow$ $\mathcal{K}$ is stably local

Proof ideas:

- consider the subterms of queries C as the elements of a partial $\mathcal{K}$-algebra F
- if C is entailed, but not locally entailed, then F cannot be weakly embedded into $\mathcal{K}$

THEOREM $\mathcal{K}$ is local $\Rightarrow$ $\mathcal{K}$ weakly embeds $\Rightarrow$ $\mathcal{K}$ is stably local

Proof ideas:

- consider the subterms of queries C as the elements of a partial $\mathcal{K}$-algebra F
- if C is entailed, but not locally entailed, then F cannot be weakly embedded into $\mathcal{K}$

Conversely if some F cannot be embedded, the function table of F gives us the defining relations of a word problem that is solvable, but not locally entailed

Embeddability vs Locality

THEOREM $\mathcal{K}$ is local $\Rightarrow$ $\mathcal{K}$ weakly embeds $\Rightarrow$ $\mathcal{K}$ is stably local

Proof ideas:

- consider the subterms of queries C as the elements of a partial $\mathcal{K}$-algebra F
- if C is entailed, but not locally entailed, then F cannot be weakly embedded into $\mathcal{K}$

Conversely if some F cannot be embedded, the function table of F gives us the defining relations of a word problem that is solvable, but not locally entailed

Technicalities are a bit messy, hence the two notions of locality

Burris' Axiomatizability Criterion

Let K be a class of relational structures.

Burris' Axiomatizability Criterion

Let K be a class of relational structures.

Full substructures of K are denoted by $S(K)$.

Burris' Axiomatizability Criterion

Let K be a class of relational structures.

Full substructures of K are denoted by $S(K)$.

Arbitrary substructures of K are denoted by $\overline{S}(K)$.

Let K be a class of relational structures.

Full substructures of K are denoted by $S(K)$.

Arbitrary substructures of K are denoted by $\overline{S}(K)$.

Observe:

$$\overline{S}(\mathcal{K}^*) = \{P^* \mid P \text{ weakly embeds into an algebra } A \in \mathcal{K}\}.$$

Burris' Axiomatizability Criterion

Let K be a class of relational structures.

Full substructures of K are denoted by $S(K)$.

Arbitrary substructures of K are denoted by $\overline{S}(K)$.

Observe:

$$\overline{S}(\mathcal{K}^*) = \{P^* \mid P \text{ weakly embeds into an algebra } A \in \mathcal{K}\}.$$

Theorem (Burris 95) Let $\mathcal{K}$ be a quasi-variety over Σ such that there is a finite set of Horn clauses H over Σ^* with $S(\mathcal{K}^*) \subseteq H \subseteq \overline{S}(\mathcal{K}^*)$. Then the uniform word problem for $\mathcal{K}$ is decidable in polynomial time.

Burris' Axiomatizability Criterion

Let K be a class of relational structures.

Full substructures of K are denoted by $S(K)$.

Arbitrary substructures of K are denoted by $\overline{S}(K)$.

Observe:

$\overline{S}(\mathcal{K}^*) = \{P^* \mid P$ weakly embeds into an algebra $A \in \mathcal{K}\}$.

THEOREM (Burris 95) Let $\mathcal{K}$ be a quasi-variety over Σ such that there is a finite set of Horn clauses H over Σ^* with $S(\mathcal{K}^*) \subseteq H \subseteq \overline{S}(\mathcal{K}^*)$. Then the uniform word problem for $\mathcal{K}$ is decidable in polynomial time.

(Call $\mathcal{K}$ (finitely) relationally axiomatizable in this case.)

Relational Axiomatizability vs Locality

THEOREM

$\mathcal{K}$ is local

$\Rightarrow$ $\mathcal{K}$ has presentation with the embeddability property

$\Rightarrow$ $\mathcal{K}$ relationally axiomatizable

$\Rightarrow$ $\mathcal{K}$ has a stably local presentation

Relational Axiomatizability vs Locality

THEOREM

$\mathcal{K}$ is local

$\Rightarrow$ $\mathcal{K}$ has presentation with the embeddability property

$\Rightarrow$ $\mathcal{K}$ relationally axiomatizable

$\Rightarrow$ $\mathcal{K}$ has a stably local presentation

Proof ideas:

- obtain a relational axiomatization from a local presentation by its relational approximation
- construct a local presentation from any relational axiomatization by turning relations back into functions

Conclusions

- seemingly different criteria are essentially equivalent
- work about locality has more to say about how to find local representations
- subterm property (+ Horn case) essential for PTIME
- elimination of the congruence axioms $x \doteq y \rightarrow f(x) \doteq f(y)$ by flattening
- finite partial algebras from relational approximation
- potential applications: combination results by amalgamation

Conclusions

- seemingly different criteria are essentially equivalent
- work about locality has more to say about how to find local representations
- subterm property (+ Horn case) essential for PTIME
- elimination of the congruence axioms $x \doteq y \rightarrow f(x) \doteq f(y)$ by flattening
- finite partial algebras from relational approximation
- potential applications: combination results by amalgamation

Conclusions

- seemingly different criteria are essentially equivalent
- work about locality has more to say about how to find local representations
- subterm property (+ Horn case) essential for PTIME
- elimination of the congruence axioms $x \doteq y \rightarrow f(x) \doteq f(y)$ by flattening
- finite partial algebras from relational approximation
- potential applications: combination results by amalgamation

Conclusions

- seemingly different criteria are essentially equivalent
- work about locality has more to say about how to find local representations
- subterm property (+ Horn case) essential for PTIME
- elimination of the congruence axioms $x \doteq y \rightarrow f(x) \doteq f(y)$ by flattening
- finite partial algebras from relational approximation
- potential applications: combination results by amalgamation

Conclusions

- seemingly different criteria are essentially equivalent
- work about locality has more to say about how to find local representations
- subterm property (+ Horn case) essential for PTIME
- elimination of the congruence axioms $x \doteq y \to f(x) \doteq f(y)$ by flattening
- finite partial algebras from relational approximation
- potential applications: combination results by amalgamation

Conclusions

- seemingly different criteria are essentially equivalent
- work about locality has more to say about how to find local representations
- subterm property (+ Horn case) essential for PTIME
- elimination of the congruence axioms $x \doteq y \rightarrow f(x) \doteq f(y)$ by flattening
- finite partial algebras from relational approximation
- potential applications: combination results by amalgamation