Symbioses Between Mathematical Logic and Computer Science

Andreas Blass

University of Michigan
Ann Arbor, MI 48109

ablass@umich.edu

Prehistory

Algorithms sought for:

- Given a Diophantine equation, is there a solution?
- Given two triangulated topological manifolds, are they homeomorphic?
- Given a presentation of a group and two elements (as words in the generators), are they equal in the group?
- Given a (first-order) sentence, is it logically valid?

The original problem was, in each case: "Find an algorithm."

Logicians enlarged the question by suggesting that one could perhaps

- define "algorthmically computable" rigorously,
- so that "is there an algorithm?" becomes a mathematical question,
- for which one might prove a negative answer.

[Post, Gödel,Herbrand, Church, Turing]

First-Order Structures

Structures for (multi-sorted) first-order logic nicely represent

- databases
- words
- arrays
- lists
- other data structures
- configurations of Turing machines
- configurations of other hardware
- any static mathematical situation
- dynamic situations too, with a sort for time.

First-order formulas describe single steps in parallel computation.

If a step is allowed to contain much parallel work but only bounded sequentiality, then its action on configurations is given by a first-order interpretation.
Parallel abstract state machines.

And conversely, first-order formulas can be evaluated by such steps.

Iteration

Repetition of simple, first-order steps leads beyond first-order logic.

The appropriate logical tool is the fixed-point construction.

Let $\gamma(P, \vec{x})$contain the k-ary predicate variable P and variables $\vec{x} = (x_1, \ldots, x_k)$.

It defines, in any structure $\mathfrak{A}$, an operator on k-ary relations

$$\Gamma(P) = \{\vec{a} \in A^k : \mathfrak{A} \models \gamma(P, \vec{a})\}.$$

If this operator is monotone, iterating it produces the **least fixed point** Γ^∞:

$$\Gamma^0 = \varnothing \qquad \Gamma^{\alpha+1} = \Gamma(\Gamma^\alpha) \qquad \Gamma^\lambda = \bigcup_{\alpha<\lambda} \Gamma^\alpha$$

for limit ordinals λ; Γ^∞ is Γ^α for all sufficiently large α.

If Γ is not monotone but inflationary, i.e., $\Gamma(P) \supseteq P$ for all P, then the iteration produces a fixed point Γ^∞, not in general the least one, called the **iterative** or inflationary fixed point.

Monotone vs. Inflationary

The least fixed point of a monotone operator is obtained not only by the standard iterative process but by any iteration that adds, at each step, to the current P, some nonempty subset of $\Gamma(P) - P$.

In the inflationary case, such an iteration yields a fixed point, but not necessarily the iterative fixed point.

Inflationary iterations must be done synchronously.
Monotone iterations can be done asynchoronously.

Nevertheless, the extensions of first-order logic by the two sorts of fixpoint operators have the same expressive power. [Gurevich, Shelah for finite structures; Kreutzer in general]

Fixed-Point Logic

Least Fixpoint Logic: Include syntax for least fixpoints of **positive** formulas $\gamma(P, \vec{x})$.

Inflationary Fixpoint Logic: Include syntax for iterative fixpoints of formulas of the form $\gamma(P, \vec{x}) \vee P(\vec{x})$.

Either of these, added to first-order logic, captures polynomial time computability on finite linearly ordered structures. [Immerman, Vardi]

The order is essential. On sets without structure, first-order logic with (either) fixed point operation cannot define "The cardinality of the set is even."

So consider fixpoint logics with counting.

Still miss some PTime computable properties, but they are more subtle.

Key to the analysis is embedding the logic in **infinitary logic** $L_{\infty,\omega}$ with finitely many variables.

Gurevich's conjecture: No logic captures polynomial time on arbitrary finite structures.

Choiceless Polynomial Time

- Input is a first-order structure
- allow "arbitrary" data structures
- prohibit arbitrary choices (or ordering)
- polynomially much (honest) work.

Formalization is abstract state machine working, in polynomial time, over $HF(\mathfrak{A})$, the universe of hereditarily finite sets over the input structure $\mathfrak{A}$. [Blass, Gurevich, Shelah]

This can't count.

In fact, it satisfies a zero-one law. [Shelah] The usual extension-axiom approach to proving 0-1 laws doesn't work here. Shelah uses stronger extension axioms.

Add counting.

Gurevich's conjecture implies that some polynomial time computable property of inputs $\mathfrak{A}$ cannot be computed in choiceless polynomial time with counting.

No example is known yet.

Two serious attempts.

Unexpected Expressibility

For bipartite graphs, the property of having a complete matching seemed undefinable in choiceless polynomial time with counting, until Shelah exhibited a very clever definition of it.

The problem of isomorphism of Cai-Fürer-Immerman graphs seemed undefinable in choiceless polynomial time with counting, until Rossman exhibited a very clever definition of it in choiceless polynomial time (without counting!).

No such definition (even with counting) is possible if one uses only hereditarily finite sets of bounded rank. [Dawar, Richerby]

These two examples go against Gurevich's conjecture, but the (very different) cleverness needed in the proofs seems to support the conjecture.

Existential Least Fixpoint Logic

- First-order logic
- plus least fixpoint operator
- minus universal quantification.

More formally,

- terms and atomic formulas as usual
- negation only of atomic formulas that begin with **negatable** predicate symbol
- $\wedge$, $\vee$, $\exists$ as usual
- simultaneous least fixpoint for **positive** predicates.

This logic arose in several contexts: Databases [Chandra, Harel], abstract computability [Aczel], Hoare logic [Blass, Gurevich].

So it seems to be a natural fragment of first-order plus least fixpoint.

Pleasant Properties and Computational Character of Existential Least Fixpoint Logic

- captures polynomial time on structures with successor
- appropriate for Hoare logic of asserted programs
- satisfaction depends on a finite part of the structure
- iterations take at most ω steps; $\Gamma^\infty = \Gamma^\omega$
- satisfaction is preserved along homomorphisms
- validity of formulas is complete Σ_1^0
- satisfiability of formulas is complete Σ_1^0
- validity of sequents $(\forall \vec{x}\,(\varphi \rightarrow \psi))$ is complete Π_2^0
- consequence relation among sequents is complete Π_1^1

Second-Order Form

Some of these follow from the fact that existential least fixpoint formulas are equivalent to second-order formulas that are **strict** $\forall_1^1$:

$$\forall \vec{P}\, \exists \vec{x}\, \varphi$$

where $\vec{P}$ are predicate (not function) variables and φ is quantifier-free.

Satisfaction of strict $\forall_1^1$ formulas depends on only a finite part of the structure.

Validity and satisfiability are complete Σ_1^0.

But a strict $\forall_1^1$ formula can define a complete co-NP property: non-3-colorability of graphs.

What more can be said about the strict $\forall_1^1$ forms of existential least fixpoint formulas?

Their quantifier-free parts are "almost" disjunctive normal forms with at most one negative literal per disjunct. (Dual to Horn formulas)

"Almost" means a simple, validity-preserving transformation converts them to this form.

The dual, satisfiability-preserving transformation seems new.

Homomorphisms and Closed Worlds

Satisfaction of existential least fixpoint formulas is preserved by homomorphisms.

By definition, homomorphisms

- commute with interpretations of function symbols,
- preserve interpretations of positive predicate symbols,
- preserve and reflect interpretations of negatable predicate symbols,

Database picture: Homomorphisms are possible developments of the database while the real world stays the same.

One could:

- become aware of more entities
- become aware of more relationships between entities, provided the relations are positive.

No new negatable relations can arise between already known entities.

If the database doesn't say $P(\vec{a})$ for negatable P, then $P(\vec{a})$ is known to be false.

More Closed Worlds

There could also be sorts that are closed in the sense that all their elements are known.

On such sorts, homomorphisms should be surjective.

Then universal quantifiers on such sorts can be added to existential fixpoint logic.

Do any nice properties persist?

Geometric Preservation

Truth values of existential fixpoint formulas are preserved by the inverse image parts of geometric morphisms of topoi.

This implies

- satisfaction depends on only a finite part of the structure
- fixed-point iterations take at most ω steps
- satisfaction is preserved along homomorphisms.

If we add universal quantification over some sort, will we still have geometric preservation?

The necessary and sufficient condition on the sort is a version of finiteness.

Open Problem: Characterize the second-order (or higher-order) formulas whose truth values are preserved by inverse images of geometric morphisms of topoi.

Fixed Point Deduction

The compactness theorem fails for fixpoint logics.

So no finitary deductive system can be sound and complete.

But there is a natural, finitary deductive system for first-order logic plus least fixpoints.

Add the following to a standard formalization of first-order logic:

- For any formula $\gamma(P, x_1, \ldots, x_k)$ where the k-ary predicate variable occurs only positively, introduce a new k-ary predicate symbol C (intended to denote the least fixpoint of the operator Γ defined by γ).
- Add the axiom

$$\gamma(C, \vec{x}) \to C(\vec{x})$$

  saying that C is closed under Γ.
- Add the axiom schema

$$\forall \vec{x} \left(\gamma(\psi(-), \vec{x}) \to \psi(\vec{x})\right) \to \forall \vec{x} \left(C(\vec{x}) \to \psi(\vec{x})\right)$$

  saying that C is least among predicates closed under Γ.

Open Problem

Assume

$$\big(\forall x \exists^{\leq 1} y \, P(x,y)\big) \rightarrow \big(\forall x \exists^{\leq 1} y \, \gamma(P,x,y)\big).$$

Can one deduce, for the corresponding closure predicate C, that

$$\forall x \exists^{\leq 1} y \, C(x,y) \; ?$$

Nondeterminacy, Alternation

Classical logic
Meaning = criterion for truth
Connectives defined by action on truth values
Deterministic computation of truth values

Intuitionistic logic
Meaning = criterion for proof
Connectives defined by action on proofs
Nondeterministic computation of truth values, or trying to produce a witness

---

Game semantics
Meaning = rules for dialog game
Connectives defined by action on games
Alternating computation of truth values, or trying to win a play of the game

Formal systems for game semantics?
Linear logic [Girard]
Computability logic [Japaridze]