Decidable Second Order Theories

G. Mints
Stanford University
after Yu. Gurevich,
Monadic Second-Order Theories
Model-theoretic logics, Springer 1985, edited by J. Barwise, F. Feferman

May 31, 2011

Language. An elementary language L augmented by sequence of quantified set variables $X, Y, \ldots$.

Atomic formulas $t \in X$.

The intended interpretation: all subsets of a structure for L.

We consider only languages where pairing is not definable.

Drop first order variables.

Example. Just two binary predicate symbols $\subseteq, \leq$.

Chain = linearly ordered set.

$\subseteq$ is the usual inclusion of sets,

$$X \leq Y :\equiv \ (\exists x \exists y X = \{x\} \& Y = \{y\} \& x \leq y)$$

Automata

Σ is a finite alphabet

A Σ-automaton: $A = (S, T, s_{in}, F)$

$T \subseteq S \times \Sigma \times S$: the transition table

$s_{in} \in S$, $F \subseteq S$: final= accepting states.

A deterministic automaton: T is a total function.

A run of A on a word $\sigma_1, \ldots \sigma_l$ in Σ:

$$s_1, \ldots, s_l$$

accepts: $s_l \in F$.

Theorem 1 (Rabin-Scott) *Indeterministic* $\rightarrow$ *deterministic*

Theorem 2 *There is an algorithm that, given an alphabet* Σ *and a* Σ*-automaton* A *decides whether* A *accepts at least one non-empty word.*

Proof. Collaps to the one-letter alphabet. Assume A is deterministic. If n is the number of states, A is purely periodic after some $i \leq n$ states. $\square$

Monadic Theory of Finite Chains

$\subseteq, SUC$.

$SUC(X, Y)$:$\equiv$ $\exists x \exists y (X = \{x\} \& Y = \{y\} \& y = suc(x))$

$x < y$:=$\equiv$

$\forall Z \; [SUC(x) \in Z \; \& \; \forall z (z \in Z \rightarrow SUC(z) \in Z)]$

A finite chain with n subsets $X_1, \ldots, X_n$: a word $Word(C, X_1, \ldots, X_n)$ of length $|C|$ in the alphabet $\Sigma_n = \{0, 1\}^n$

Suppose $C = \{2, 3\}, X_1 = \emptyset, X_n = \{2\}$.

$$\begin{array}{cccc} & X_1 & \ldots & X_n \\ 2 & 0 & \ldots & 1 \\ 3 & 0 & \ldots & 0 \end{array}$$

Theorem 3 *There is an algorithm that, given n and a Σ_n-automaton A, constructs a formula $\phi(X_1, \ldots, X_n)$ in the monadic language of one successor such that for every finite chain C and any subsets $X_1, \ldots, X_n$ of C we have that*

$$C \models \phi(X_1, \ldots, X_n)$$

iff A accepts $Word(C, X_1, \ldots, X_n)$.

□

Theorem 4 *There is an algorithm that, given a formula $\phi(X_1, \ldots, X_n)$ in the monadic language of one successor constructs a Σ_n-automaton A such that for every finite chain C and any subsets $X_1, \ldots, X_n$ of C we have that*

$$C \models \phi(X_1, \ldots, X_n)$$

iff A accepts $Word(C, X_1, \ldots, X_n)$.

□

A kind of normal form theorem.

Theorem 5 *The monadic theory of finite chains is decidable.*

Proof. Given a sentence ϕ, find an appropriate automaton, check whether it accepts at least one non-empty word.

Monadic Theory of ω

Language: $\subseteq, SUC(X, Y)$.

$X, Y, \ldots$ range over subsets of ω, $\leq$ is definable as before.

A *sequential* Σ-automaton:

$A = (S, T, s_{in}, F)$, F is the set of final collections of states. Non-deterministic.

A *run* of A on a sequence $\sigma_1, \sigma_2 \ldots$

is a sequence $s_1, s_2, \ldots$ of states such that

$(s_{in}, \sigma_1, s_1) \in T$ and every $(s_i, \sigma_{i+1}, s_{i+1}) \in T$.

It is an accepting run if

$\{s : \; s_n = s$ for infinitely many $n\} \in \; T$.

A accepts a sequence if there is an accepting run of A on this sequence.

Theorem 6 *There is an algorithm that, given an alphabet Σ and a sequential Σ-automaton A, constructs a deterministic sequential*

Σ-automaton accepting exactly the sequences accepted by A.

McNaughton, 1966.

Theorem 7 *There is an algorithm that, given an alphabet Σ and a sequential Σ-automaton A, decides whether A accepts at least one sequence.*

Proof. Again by periodicity. $\square$

Subsets $X_1, \ldots X_n$ of ω form a sequence

$SEQ(X_1, \ldots, X_n)$ in the alphabet Σ_n.

Theorem 8 *There is an algorithm that, given n and a Σ_n-automaton A, constructs a formula $\phi(X_1, \ldots, X_n)$ in the monadic language of one successor such that for any subsets $X_1, \ldots, X_n$ of ω we have that*

$$\omega \models \phi(X_1, \ldots, X_n) \text{ iff } A \text{ accepts } SEQ(X_1, \ldots, X_n) \ .$$

□

Theorem 9 *There is an algorithm that, given a formula* $\phi(X_1, \ldots, X_n)$ *in the monadic language of one successor constructs a* Σ_n*-automaton* A *such that for every finite chain* C *and any subsets* $X_1, \ldots, X_n$ *of* ω *we have that*

$$\omega \models \phi(X_1, \ldots, X_n) \text{ iff } A \text{ accepts } SEQ(X_1, \ldots, X_n) \ .$$

□

Theorem 10 *The monadic theory of* ω *is decidable.*

Monadic Theory of the Binary Tree: S2S.

The binary tree: the set $\{l,r\}^*$ of all words in the alphabet $\{l,r\}$.

$xl,\ xr$ are successors of x.

The *monadic language of two succesors* is (formally) the first-order language with binary predicates $\subseteq, Left, Right$.

$Left(X,Y) :\equiv X = \{x\},\ Y = \{xl\}$ for some word x.

The relations "x is the initial segment of y", "$x \prec y$ lexicographically" are easily expressible. Rabin [1969] interpreted monadic theories of 3,4, etc. successors, ω successors and much more.

Σ-tree: a mapping V from the binary tree to Σ.

A Σ-tree automaton $A = (S, T, T_{in}, F)$

$T \subseteq S \times \{l, r\} \times \Sigma \times S$

$T_{in} \subseteq \Sigma \times S$: initial state table

F: the set of *final collections of states*.

A game $\Gamma(A, V)$ between A and the Pathfinder

A chooses	P chooses
s_0	d_1
s_1	d_2
$\dots$	$\dots$

$s_n \in S, \ d_n \in \{l, r\}$

$(V(e), s_0) \in T_{in}$,

$(s_n, d_{n+1}, V(d_1 \dots d_{n+1}), s_{n+1}) \in T$.

Additional state FAILURE: a transition to it is always possible, but not to any other state.

{FAILURE} is not in a final collection.

A wins a play $s_0 d_1 s_1 d_2 \ldots$ if

$$\{s \in S : \; s_n = s \text{ for } \infty \; n\} \; \in \; F$$

Otherwise P wins.

A accepts a tree V if it has a winning strategy in $\Gamma(A, V)$. Otherwise A rejects V.

Theorem 11 *There is an algorithm that, given an alphabet* Σ *and a tree* Σ*-automaton* A*, decides whether* A *accepts at least one* σ*-tree.*

Proof. Again by periodicity. □

Subsets $X_1, \ldots X_n$ of the binary tree form a Σ_n-tree

$TREE(X_1, \ldots, X_n)$.

Theorem 12 *There is an algorithm that, given* n *and a* Σ_n*-automaton* A*, constructs a formula* $\phi(X_1, \ldots, X_n)$ *in the monadic language of two successors such that for any subsets* $X_1, \ldots, X_n$ *of the binary tree*

$$\{l, r\}^* \models \phi(X_1, \ldots, X_n)$$

iff A *accepts* $TREE(X_1, \ldots, X_n)$.

□

Theorem 13 *There is an algorithm that, given a formula* $\phi(X_1, \ldots, X_n)$ *in the monadic language of two successors constructs a* Σ_n*-automaton* A *such that for any subsets* $X_1, \ldots, X_n$ *of the binary tree*

$$\{l, r\}^* \models \phi(X_1, \ldots, X_n)$$

$$\textit{iff } A \textit{ accepts } TREE(X_1, \ldots, X_n) \ .$$

$\Box$

Theorem 14 *The monadic theory of the binary tree is decidable.*

Proof. As before, but the complementation theorem requires a complicated argument (simplified by Gurevich and Harrigton) based on Ramsey Theorem. $\Box$

Theories decidable by interpretyation in S2S.

Many (including ω) successors.

The first-order theory of closed (and F_σ) subsets of the real line;

The second-order theory of countable linearly ordered sets;

The second-order theory of countable well-ordered sets;

The theory of countable Boolean algebra with quantification over ideals;

The weak second-order theory of a unary function,

etc.