Algebra Meets Finite Model Theory

Howard Straubing
Boston College

Journées d'Informatique Fondamentale de Paris Diderot
April 26, 2013

Outline of this talk

- 0. Prelude: What good are semigroups?
- 1. Four decades of research on the application of the algebraic theory of semigroups to problems about the expressive power of logics on words.
- 2. Efforts (failures? challenges?) to extend the reach of these techniques beyond regular languages of finite words: (a) boolean circuit complexity (b) logics on trees.

Part 1: Weeds

A question on my oral qualifying exam in Algebra, 1975

How many groups of order 45 are there?

A question on my oral qualifying exam in Algebra, 1975

How many groups of order 45 are there?

Answer: 2, both abelian. (I Googled it.)

Semigroups

A **semigroup** is a set together with an associative multiplication. (A **monoid** is a semigroup with an identity element.)

Semigroups suffer from an image problem: Even the name makes it sound like it is something trying hard to be a group but not succeeding.

Theorem: Almost all finite semigroups are *garbage*.

(Kleitman, Rothchild, Spencer)

- $s(n)$: number of semigroups of order n.
- $t(n)$: number of semigroups of order n satisfying identity $xyz = 0$ where $0 \cdot x = 0 = x \cdot 0$ for all elements x.
- Then
$$\lim_{n \to \infty} \frac{t(n)}{s(n)} = 1.$$

“Finite group theory is like an exotic garden, where everything that grows is a beautiful flower. Finite semigroup theory is more like a yard full of weeds.”

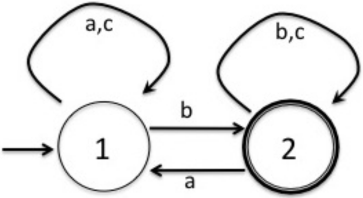
–*John Rhodes*

This is not a defect of semigroup theory! It is a result of not looking at semigroups the right way or asking the right questions about them.

Part 2: Where do Semigroups Appear in Natre?

Finite automata

Minimal automaton accepting the set of strings corresponding to the regular expression $(a+b+c)^*bc^*$



Syntactic monoid

Each word w in $\{a, b, c\}^*$ induces a function $[w]$ from $\{1, 2\}$ to itself.

$$\begin{array}{rlll} c^* : & & 1 \mapsto 1 & 2 \mapsto 2 \\ (a+b+c)^* bc^* : & & 1 \mapsto 2, & 2 \mapsto 2 \\ (a+b+c)^* ac^* : & & 1 \mapsto 1, & 2 \mapsto 1 \end{array}$$

This is a 3-element monoid U_2 with left-to-right function composition as the operation. ($[u][v] = [uv]$.)

The pair $(\{1, 2\}, U_2)$ is a **transformation monoid.**

$M(L)$: **syntactic monoid** of $L \subseteq A^*$, transition monoid of minimal automaton of L.

Big Idea

Structure of the syntactic monoid of a language tells us something about definability of the language in various logics.

Part 3. The Theorem of McNaughton-Papert and Schützenberger, and its Progeny

$FO[<]$

Variables represent positions in a string over finite alphabet A.

Atomic formulas: $x < y$ means position x is strictly to the left of position y, $Q_a x$ means position x contains the letter $a \in A$.

Boolean operations and quantifiers have their usual meaning.

Our example language consists of strings over $A = \{a, b, c\}$ containing a b with no a's after it:

$$\exists x (Q_b x \wedge \forall y (Q_a y \rightarrow y < x)).$$

What can you say in first-order logic?

What properties of words can be defined in $FO[<]$?

Every language definable in $FO[<]$ is regular (since $FO[<] \subseteq MSO$). Is there some way of determining from an automaton whether the language it accepts is first-order definable?

A necessary condition for first-order definability

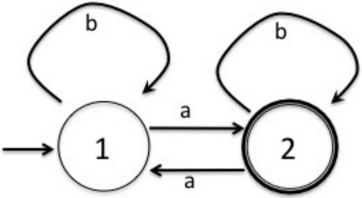
Let $w \in A^*$. Second player ('duplicator') wins the n-round Ehrenfeucht-Fraïssé game for $FO[<]$ in $(w^{2^{n-1}}, w^{2^{n-1}+1})$.

So $w^{2^{n-1}}$ and $w^{2^{n-1}+1}$ must induce the same state transitions in any automaton that recognizes a language L definable by a $FO[<]$ formula of quantifer depth $\leq n$.

Thus if L is definable in $FO[<]$ then $M(L)$ satisfies the identity $x^k = x^{k+1}$ for all sufficiently large k.

Equivalently $M(L)$ must be **aperiodic** : it contains no nontrivial groups.

For instance, the set of strings over $\{a, b\}$ with an odd number of a's is not definable in $FO[<]$, because its syntactic monoid is the group of permutations of the two states of the minimal automaton.



The right way to think about semigroups

Such equationally-defined classes (‘satisfies $x^k = x^{k+1}$ for k sufficiently large’) are **pseudovarieties of finite monoids**–closed under finite direct products, quotients and submonoids.

The right questions to ask usually involve the structure of, and relations between, such varieties.

But there’s more!

Wreath product of transformation monoids (cascade composition of automata)

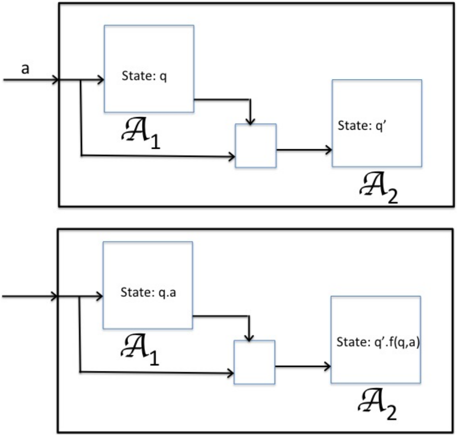


Figure : Wreath product before and after application of input a : $(q, q') \cdot a = (q \cdot a, q' \cdot f(q, a))$.

Krohn-Rhodes Theorem: Deep structure of aperiodic semigroups

If $M(L)$ is aperiodic, then L is recognized by a wreath product of copies of our 2-state, 3-element transformation monoid U_2.

This is typical of the kinds of ‘decomposition’ results of this theory: we do not really break $M(L)$ into smaller pieces, we **cover** it by something that can be broken into smaller pieces.

McNaughton-Papert/Schützenberger Theorem: Aperiodicity is Necessary and Sufficient for First-order Definability

It follows from an induction on the number of factors in the cascade that if $M(L)$ is aperiodic, then L is definable by a formula of $FO[<]$.

This is a different kind of model-theoretic result: We have an **effective criterion for determining whether the behavior of an automaton is first-order definable,** based on computing the transition monoid of the reduced automaton and verifying identities.

A flood of results, beginning in the 1970's and continuing to the present, use the decomposition theory for finite semigroups to obtain such effective characterizations of logically defined classes of regular languages.

First-order logic with successor (Beauquier and Pin 1991, building on Thérien and Weis 1986)

FO[succ]: Atomic formulas $x < y$ replaced by $y = x + 1$.

$L \subseteq A^*$ definable if and only if $M(L)$ is aperiodic, and for all $e, f, s, t, u \in M(L)$ induced by nonempty words, with $e^2 = e, f^2 = f$,

$$esfteuf = euftesf$$

Associated decomposition theorem: Satisfaction of these identities is equivalent to recognition by a wreath product of an aperiodic and commutative monoid with a ‘definite’ semigroup: One that satisfies $se = e$ for all idempotents e.

Modular Quantifiers (Straubing, Thérien, Thomas 1988)

$(FO + MOD)[<]$ is $FO[<]$ supplemented by quantifiers $\exists^{r \bmod q}$ that say 'there are exactly r mod q positions such that...'

For example $\exists^{1 \bmod 2} x Q_a x$ defines the set of strings with an odd number of occurrences of a.

Theorem: L is definable in $(FO + MOD)[<]$ if and only if every group in $M(L)$ is solvable.

Associated decomposition theorem: Krohn-Rhodes Theorem–monoids with only solvable groups are the wreath product closure of U_2 and the cyclic groups.

Modular Quantifiers (Straubing, Thérien, Thomas 1988)

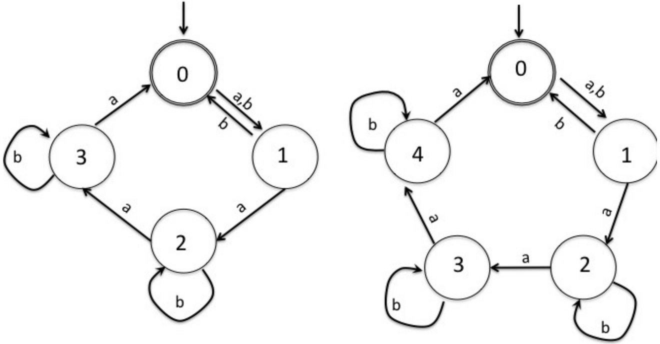


Figure : The behavior of the automaton on the left is definable using only modular quantifiers of modulus 2 and 3; the one on the right is not definable with any combination of ordinary and modular quantifiers.

Fragments of linear temporal logic (Thérien and Wilke 1996)

Let $L \subseteq A^*$. The following are equivalent:

- L is definable in the fragment of linear temporal logic using only strict past and strict future operators.
- L is definable by a formula of $FO[<]$ using only two variables.
- L is definable by both a Σ_2 formula of $FO[<]$ and by a Π_2 formula.
- $M(L) \in \mathbf{DA}$: it satisfies the identites

$$xx^\omega = x^\omega, (xyz)^\omega y(xyz)^\omega = (xyz)^\omega$$

where x^ω denotes the unique idempotent power of x.

Associated decomposition theorem: **DA** is the 'weak block-product closure' of the idempotent and commutative monoids.

This algebraic approach is a powerful tool for the study of **regular languages of finite words.** Can we move beyond this?

Part 4: Connection with Bounded-Depth Circuits

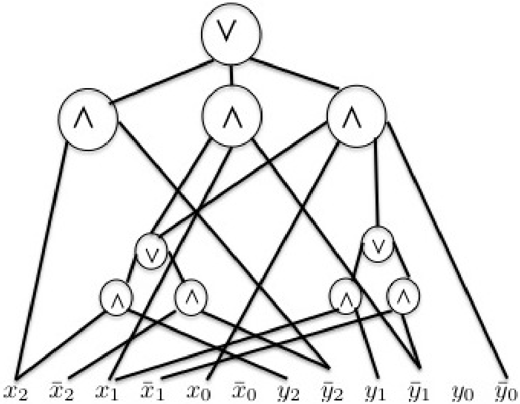
Comparison of two integers in binary

$x_{n-1} \cdots x_0 y_{n-1} \cdots y_0 \in \{0,1\}^{2n}$.

$(x_{n-1} \cdots x_0)_2 > (y_{n-1} \cdots y_0)_2$ if and only if

$$\bigvee_{j=1}^{n} \Big(x_{n-j} \wedge \bar{y}_{n-j} \wedge \bigwedge_{i=j+1}^{n} (x_{n-i} = y_{n-i}) \Big) = 1.$$

AC^0



AC^0: family of languages recognized by bounded-depth, polynomial-size families of circuits with unbounded fan-in AND and OR gates.

Interpretation in terms of nonuniform automata

If we read **pairs** of input symbols in the order

$$(x_0, y_0), (x_1, y_1), \ldots, (x_{n-1}, y_{n-1}),$$

then this is recognized by the two-state U_2-automaton.

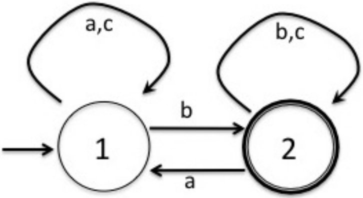


Figure : $(0, 0), (1, 1)$ encoded as c, $(0, 1)$ as a, $(1, 0)$ as b

...and in terms of first-order logic

From the first-order sentence for the behavior of the U_2-automaton

$$\exists x(Q_b x \wedge \forall y(Q_a y \rightarrow y < x)).$$

we obtain a sentence for the comparison language

$$\exists z_1 \exists z_2(\mathrm{C}(z_1, z_2) \wedge Q_1 z_1 \wedge Q_0 z_2 \quad \wedge \\ \forall z_3 \forall z_4((\mathrm{C}(z_3, z_4) \wedge Q_0 z_3 \wedge Q_1 z_4) \rightarrow z_1 < z_3))$$

where $\mathrm{C}(z, z')$ means $z' = z + \frac{\text{length}}{2}$.

Circuits, automata, logic

Theorem (Immerman for equivalence of first two items, Barrington and Thérien for the third): The following are equivalent:

- $L \in AC^0$.
- L definable in $FO[\mathcal{N}]$ (first-order logic with *unrestricted* numerical predicates).
- L recognized by nonuniform **aperiodic** finite automaton reading k-tuples of input bits.

Lower bounds and their interpretation in terms of automata and logic

- Theorem (Furst, Saxe, Sipser). Let $q > 1$. The set of bit strings in which the number of 1's is divisible by q is not in AC^0.
- Theorem (Barrington, Compton, Straubing, Thérien).

$$FO[\mathcal{N}] \cap \text{Regular} \quad \text{languages} = FO[<, x \equiv 0 \pmod{q}].$$

- Circuit lower bounds are equivalent to statements about the definability of **regular languages in first-order logic!** Can we use this observation to give a different proof of these bounds?

Modular gates and ACC^0.

If we add a new gate type that determines if the number of 1's on the input is divisible by n, we get the class $ACC^0[n]$. $ACC^0 = \cup_{n>0} ACC^0[n]$.

$ACC^0 = (FO + MOD)[\mathcal{N}] =$ behavior of nonuniform automata over monoids that contain only solvable groups.

Huge open problem in circuit complexity: Can we solve the word problem for a non-solvable group in ACC^0? (Separation of ACC^0 from NC^1.)

The logic/automata interpretation

Huge open problem in circuit complexity: Can we solve the word problem for a non-solvable group in ACC^0?

Equivalent formulation: Does the following hold?

$$(FO + MOD)[\mathcal{N}] \cap \text{Regular} \quad \text{languages} = (FO + MOD)[<]$$

Can such problems be approached using our algebraic tools?

Baby steps

Using a combination of Ramsey-style combinatorics and semigroup theory we can prove:

(Barrington-Straubing, 1995):

$(\mathit{FO}{+}\mathit{MOD})[\mathcal{N}^{\text{monadic}}, <] \cap \text{Regular} \quad \text{languages} = (\mathit{FO}{+}\mathit{MOD})[<]$

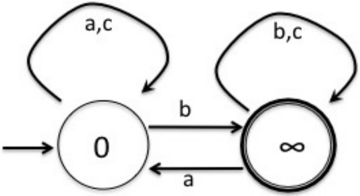
(Roy-Straubing, 2007):

$$(\mathit{FO} + \mathit{MOD})[+] \cap \text{Regular} \quad \text{languages} = (\mathit{FO} + \mathit{MOD})[<]$$

Part 5: Trees and Forests

Forest algebras-(Bojanczyk-Walukiewicz 2008)

If we put an additive structure ($0+\infty=\infty$) on our U_2 automaton, we can use it as a bottom-up tree automaton, to read **labeled forests** instead of just words.



Our transformation monoid (Q, M) has become a **forest algebra** (H, V), where the set H of states has a monoid structure (which we write additively).

The set of forests accepted consists of all those that contain a node labeled *b* that has no ancestor labeled *a*.

$$\exists x(Q_b x \wedge \forall y(Q_a y \rightarrow \neg(y \prec x))).$$

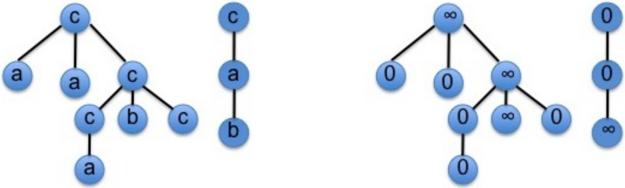


Figure : The value is $0 + \infty = \infty$

- There is a syntactic forest algebra (H_L, V_L) associated to each collection L of forests (essentially the minimal acceptor).
- Can one obtain effective characterizations of properties definable in $FO[\prec]$? or in related logics such as CTL?
- We can prove abstractly that these properties are indeed **algebraic** and depend only on the structure of the syntactic forest algebra.

Decompositions and necessary conditions (Bojanczyk-Straubing-Walukiewicz, 2012)

- Theorem: L is definable in CTL if and only if it is recognized by a wreath product of copies of U_2 (shades of Krohn-Rhodes!)
- Theorem: L is definable in $FO[\prec]$ if and only if L is recognized by a wreath product of forest algebras (H, V) that satisfy the identities

  $$gv + hv = (g + h)v + 0v$$

  for all $g, h \in H$, $v \in V$.

Decompositions and necessary conditions (Bojanczyk-Straubing-Walukiewicz, 2012)

- We can use these characterizations to derive **effective, algebraic necessary conditions** for definability in these logics, and prove that certain properties are not definable.
- $$CTL \subsetneq FO[\prec] \subsetneq \{L : V_L \quad \text{aperiodic}\}$$
- The conditions we derive are not sufficient! Can we duplicate the successes of the theory for finite words?

What I didn’t talk about

- Characterizing the quantifier alternation depth in $FO[<]$: The long-open dot-depth problem.
- Promising successes for forest algebras: First-order logic with successor (Benedikt-Segoufin), piecewise-testable forest languages (Bojanczyk-Segoufin-Straubing),...
- A topological approach to recognizability (with possible applications to circuit complexity)? (Gehrke-Grigorieff-Pin).
- An algebraic-logical theory for regular cost functions (Colcombet)

Thank you