Machines, Models, Monoids, and Modal logic

Sam van Gool

University of Amsterdam and City College of New York

September 2017
Tbilisi Symposium on Language, Logic and Computation
Lagodekhi, Georgia

Outline

1. Part I: Formal Languages, Automata, and Algebra
2. Part II: Duality and Varieties of Monoids
3. Part III: Profiniteness, Pointlikes, and the Future

Outline

1. Part I: Formal Languages, Automata, and Algebra
2. Part II: Duality and Varieties of Monoids
3. Part III: Profiniteness, Pointlikes, and the Future

Outline Part I

1. What is a formal language?
   - Alphabets and words
   - Formal languages
2. How to describe a formal language?
   - Automata
   - Logic
   - (Open) Problems
3. How to understand formal languages?
   - Boolean algebras with operators
   - Model theory
   - Monoids

Formal language theory

- A mathematical setting for analyzing computational problems.
- Or: ... formal grammars.
- All definitions are elementary.
- Many problems are difficult, interesting, and often open.

1. What is a formal language?
   - Alphabets and words
   - Formal languages

Alphabets and words

- An *alphabet* is a finite set of symbols, Σ.
- A finite Σ-word is a finite sequence of elements of Σ.

Alphabets and words

Examples

- If $\Sigma = \{b, l, i, s, t, B, L, I, S, T\}$ then three examples of (distinct!) Σ-words are: tbilisi, Tbilisi, and TBILISI.
- If $\Sigma = \{\texttt{enter\_coin}, \texttt{push\_cola}, \texttt{push\_water}\}$ then three examples of Σ-words are: $(\texttt{enter\_coin}, \texttt{push\_cola})$, $(\texttt{push\_cola}, \texttt{push\_water}, \texttt{push\_cola})$, and $(\texttt{push\_cola}, \texttt{push\_cola}, \texttt{push\_cola}, \texttt{push\_cola}, \texttt{push\_cola})$. The last one can be briefly denoted as: $\texttt{push\_cola}^5$.
- The empty word, ϵ, is a word in any alphabet.

Formal Languages

- *Notation:* Σ^* is the set of all Σ-words.
- A (formal) Σ-*language* is a subset L of Σ^*.

Examples

- The *empty language*, $\emptyset$.
- The language containing only the empty word, $\{\epsilon\}$.
- The set of all Σ-words, Σ^*.
- The set of non-empty words is a language, $\Sigma^+ = \Sigma^* \setminus \{\epsilon\}$.

Formal Languages

Examples

- Let Σ be the set of all lower-case letters, capital letters, numbers, and the symbols $\texttt{!}, \texttt{@}, \texttt{\#}, \texttt{\$}, \ast, \texttt{(}, \texttt{)},$ and $\texttt{\%}$. An example of a Σ-language is

$$\texttt{PW} = \{w \in \Sigma^* \mid w \text{ is at least 8 characters long and contains at least one letter, one number, and one special symbol}\}.$$

- Let $\Sigma = \{\texttt{enter\_coin}, \texttt{push\_cola}, \texttt{push\_water}\}$. An example of a Σ-language is

$$\texttt{BUY} = \{w \in \Sigma^* \mid w \text{ contains an occurrence of } \texttt{enter\_coin} \text{ before an occurrence of } \texttt{push\_cola} \text{ or } \texttt{push\_water}\}.$$

- Let $\Sigma = \{0, 1\}$. Three examples of $\{0, 1\}$-languages are:

$\texttt{FACTOR01} = \{w \in \{0, 1\}^* \mid w \text{ contains `01' as a factor}\}$.

$\texttt{EVENONES} = \{w \in \{0, 1\}^* \mid \text{ the number of 1's in } w \text{ is even}\}$.

$\texttt{NON1} = \{0^n 1^n \mid n \geq 0\}$.

2. How to describe a formal language?
    - Automata
    - Logic
    - (Open) Problems

Describing formal languages

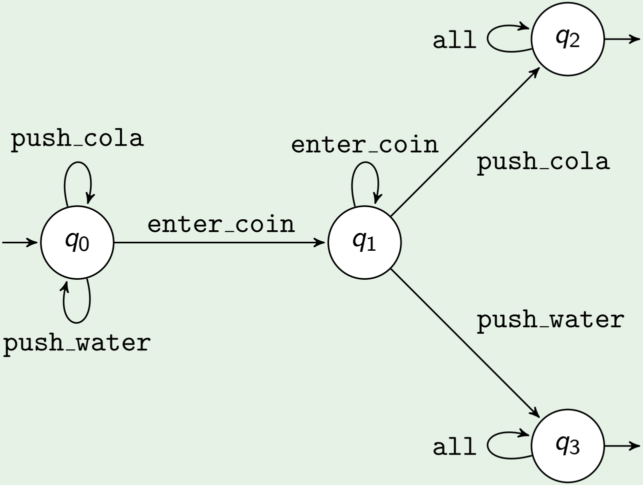
- Formal grammars
- Machines
- Logic

In this tutorial, we will focus on the last two, and we will mostly restrict to regular languages.

Automata

Examples

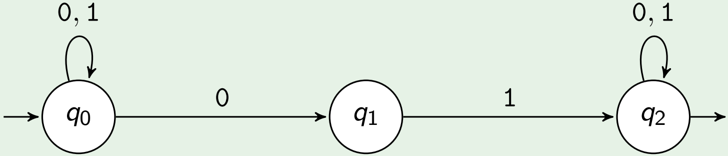
An automaton for the language BUY.



Automata

Examples

An automaton for the language
$\mathtt{FACTOR01} = \{w \in \{0,1\}^* \mid w \text{ contains `01' as a factor}\}$.



Automata

- An *automaton* is a tuple $\mathcal{A} = (Q, \Sigma, \delta)$, where
  - Q is a finite set of *states*,
  - Σ is a finite *alphabet*,
  - $\delta \colon Q \times \Sigma \to \mathcal{P}(Q)$ is a *transition function*.
- *Notation:* $q \xrightarrow{a} q'$ means: q is a state in Q, a is a letter in Σ, and q' is a state in $\delta(q, a)$.
- Pick two sets of states, I and F, in Q.
- A word $a_1 \dots a_n \in \Sigma^*$ is *accepted by the automaton $\mathcal{A}$ with initial states I and final states F* if there exists a path $q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} q_n$ such that $q_0 \in I$ and $q_n \in F$.
- The language $L_{\mathcal{A},I,F}$ of all accepted words is called the language *recognized* by $\mathcal{A}$ with initial states I and final states F.
- Full technical name: *non-deterministic finite automaton* (NFA).
- *Deterministic* (DFA): $\delta \colon Q \times \Sigma \to Q$.

Regular languages

- A language L is called *regular* if there exists an NFA that recognizes it.

Fact

There exists a non-deterministic finite automaton that recognizes L if, and only if, there exists a deterministic finite automaton that recognizes L.

Exercises

1. Describe an automaton that recognizes `EVENONES`.
2. Describe an automaton that recognizes `PW`.
3. Describe a *deterministic* automaton that recognizes `FACTOR01`.
4. (*) Is it possible to find an automaton that recognizes `NON1`? If no, explain why not.

Logic

Examples

- The language BUY of action sequences for buying has logic description

  $$\exists x \exists y\, [x < y \wedge \mathtt{enter\_coin}(x) \wedge (\mathtt{push\_cola}(y) \vee \mathtt{push\_water}(y))]\,.$$

- The language EVENLENGTH of $\{0,1\}$-words of even length has logic description

  $$\mathtt{empty} \vee \exists P\, [P(\mathtt{first}) \wedge \neg P(\mathtt{last}) \wedge \forall x \neq \mathtt{last}\, (P(x) \leftrightarrow \neg P(\mathtt{S}(x)))]\,.$$

Logic

- Syntax:
  - Basic propositional connectives: $\wedge$, $\neg$.
  - Quantification over first-order variables x, y, $\ldots$ and monadic second-order variables P, Q, $\ldots$.
  - Atomic formulas: $x < y$, $P \subseteq Q$, $P(x)$, $\mathrm{a}(x)$ for $a \in \Sigma$.
- Semantics: view a word $w = a_1 \ldots a_n$ as a *structure* W, i.e.,
  - The underlying set of W is $\{1, \ldots, n\}$.
  - The natural linear order $<^W$ interprets the binary predicate $<$.
  - For every letter $a \in \Sigma$, a^W is the set of positions i where $a_i = a$.
- For a sentence φ, $L_\varphi = \{w \in \Sigma^* \mid w \models \varphi\}$.
- Shortcuts such as $\mathtt{S}(x)$, `first`, `last`, `empty`, ... are definable.
- This is Monadic Second Order (MSO) Logic.
- First Order (FO) Logic is obtained by disallowing second order variables and second order quantifiers.

Exercises

1. Describe the language FACTOR01 with MSO logic, or FO logic if possible.
2. Describe the language PW with MSO logic, or FO logic if possible.
3. Describe the language EVENONES with MSO logic, or FO logic if possible.
4. If you think it is impossible to find an FO logic definition in (1)–(3), explain why.
5. What is the lowest possible quantifier depth you need to describe PW and EVENONES? (*) Can you prove it?
6. (*) Is it possible to describe the language NON1 with an MSO formula? If no, why not?

Logic and automata

Theorem (Büchi 1960)

Let $L \subseteq \Sigma^*$ be a language. Then L is regular if, and only if, L is definable in MSO logic.

Proof ingredients.

- The behavior of any automaton can be 'described' in MSO logic.
- MSO logic can be simulated by automata. □

From here on, regular = MSO-definable.

Problems

- Given an automaton, decide if it accepts any words?
- Given a regular language, decide if it is FO-definable?
- Given an FO-definable language, decide if it is definable in FO_k, i.e., FO logic of quantifier depth $\leq k$?
- Given two regular languages, decide if they are *separable* by an FO-definable language?
    - A language M *separates* L_1 from L_2 if $L_1 \subseteq M$ and $L_2 \cap M = \emptyset$.
- Given two regular languages, decide if they are *separable* by an FO_k-definable language?
- …

Problems

- Given an automaton, decide if it accepts any words? See next slides
- Given a regular language, decide if it is FO-definable? See Part II
- Given an FO-definable language, decide if it is definable in FO_k, i.e., FO logic of quantifier depth $\leq k$? Open for $k \geq 3$
- Given two regular languages, decide if they are *separable* by an FO-definable language? See Part III
  - A language M *separates* L_1 from L_2 if $L_1 \subseteq M$ and $L_2 \cap M = \emptyset$.
- Given two regular languages, decide if they are *separable* by an FO_k-definable language? Open for $k \geq 3$
- …

3. How to understand formal languages?
   - Boolean algebras with operators
   - Model theory
   - Monoids

Boolean algebras with operators

- The set of all Σ-languages, $\mathcal{P}(\Sigma^*)$, is a Boolean algebra with operations $\cup$ (union) and $()^{\mathrm{c}}$ (complement).
- For any letter $a \in \Sigma$, the function

$$L \mapsto a^{-1}L = \{w \in \Sigma^* \mid aw \in L\}$$

is an *endomorphism* of the Boolean algebra, and so is

$$L \mapsto La^{-1} = \{w \in \Sigma^* \mid wa \in L\}.$$

Fact

- *If L_1, L_2 are regular, then $L_1 \cup L_2$ is regular.*
- *If L is regular, then L^{c} is regular.*
- *If L is regular, then $a^{-1}L$ and La^{-1} are regular.*

Quotient operators shift initial and final states

Proof of last item.

- Suppose that $\mathcal{A}$ is an NFA that recognizes L with initial states I and final states F.
- Then $a^{-1}L$ is recognized by $\mathcal{A}$ with final states F and initial states Ia, i.e., the set of states q which admit a transition $q_0 \xrightarrow{a} q$ for some $q_0 \in I$.
- Also, La^{-1} is recognized by $\mathcal{A}$ with initial states I and final states $a^{-1}F$, i.e., the set of states q which admit a transition $q \xrightarrow{a} q_F$ for some $q_F \in F$. □

Corollary

If L is regular, then the set $\{w^{-1}L, Lw^{-1} \mid w \in \Sigma^*\}$ is finite.

Boolean algebras with operators

- A Boolean subalgebra $B \leq \mathcal{P}(\Sigma^*)$ is *closed* if, for every L in B and a in Σ, both $a^{-1}L$ and La^{-1} are in B.
- The set of *regular* Σ -languages, $\mathrm{Reg}(\Sigma^*)$, is a closed subalgebra of $\mathcal{P}(\Sigma^*)$.
- For any automaton $\mathcal{A} = (Q, \Sigma, \delta)$, the set of Σ-languages which $\mathcal{A}$ can recognize is a finite closed subalgebra of $\mathrm{Reg}(\Sigma^*)$.
- Any Σ-language L *generates* a closed subalgebra, $B(L)$, i.e., the *smallest* closed subalgebra containing L.

Proposition

A language $L \in \mathcal{P}(\Sigma^*)$ is regular if, and only if, $B(L)$ is finite.

Exercises

1. Describe the closed subalgebra generated by the $\{0,1\}$-language $\texttt{EVENLENGTH}$.
2. Let $S \subseteq \mathbb{N}$. Describe the closed subalgebra generated by the $\{1\}$-language $\texttt{LENGTH}_S$ of $\{1\}$-words of length S.
3. (*) When is the algebra in (2) finite?

Model theory

- Let T_Σ be the MSO theory of finite Σ-words, i.e., the set of MSO sentences that are true in all finite Σ-words.
- Let $\mathcal{L}(T_\Sigma)$ be the Lindenbaum algebra of T, i.e., the algebra of MSO-sentences up to T_Σ-equivalence. This is a Boolean algebra.
- To any $[\varphi]_{T_\Sigma}$ in $\mathcal{L}(T_\Sigma)$, associate the regular language, $L(\varphi)$, described by φ.
- This assignment is a well-defined isomorphism between $\mathcal{L}(T_\Sigma)$ and $\mathrm{Reg}(\Sigma^*)$.
- Exercise: (*) Describe the operators $L \mapsto a^{-1}L$ and $L \mapsto La^{-1}$ directly on the Lindenbaum algebra $\mathcal{L}(T_\Sigma)$.
- Under this isomorphism, the subalgebra of FO-sentences corresponds to a subalgebra of $\mathrm{Reg}(\Sigma^*)$. Which? See Part II

Semigroups and monoids

- A *semigroup* is a pair $(S, \cdot)$, where $\cdot$ is an associative operation, i.e., $x \cdot (y \cdot z) = (x \cdot y) \cdot z$ for all x, y, z in S.
- A *monoid* is a semigroup that contains an *identity element*, 1, i.e., $1 \cdot x = x \cdot 1$ for all x in S.

Monoids

Examples

- The set Σ^*, with multiplication $u \cdot v := uv$.
- For any set P, the set of functions from P to itself, $(P \to P)$, with multiplication $f \cdot g := f \circ g$.
- In particular, an NFA $\mathcal{A} = (Q, \Sigma, \delta)$ gives, for every $a \in \Sigma$, a function $\Diamond_a$ in $(\mathcal{P}(Q) \to \mathcal{P}(Q))$, defined by

$$\Diamond_a(R) := \{q \mid q \xrightarrow{a} q' \text{ for some } q' \in R\}.$$

Exercises

1. Show that Σ^* is a monoid.
2. Show that $(P \to P)$ is a monoid.
3. Show that Σ^* is the *free* monoid on Σ, i.e., that for any monoid M and any function $f \colon \Sigma \to M$, there is a unique homomorphism $\bar{f} \colon \Sigma^* \to M$ extending f.
4. Applying (3) to the function $\Diamond \colon \Sigma \to (\mathcal{P}(Q) \to \mathcal{P}(Q))$, give an explicit description of the function $\bar{\Diamond} \colon \Sigma^* \to (\mathcal{P}(Q) \to \mathcal{P}(Q))$.
5. (*) Show that $\mathcal{A}$ with initial states I and final states F accepts a word $w \in \Sigma^*$ if, and only if, $I \cap \bar{\Diamond}_w(F) \neq \emptyset$.

References for Part I

- An accessible textbook introduction to the field:

  P. Linz. *An introduction to formal languages and automata*. 5th ed. Jones & Bartlett, 2012

- A more advanced, but very readable introduction to logic on words:

  H. Straubing. *Finite automata, formal logic, and circuit complexity*. Progress in Theoretical Computer Science. Birkhäuser Boston Inc., Boston, 1994

- Our recent work on applications of model theory: (see also part II)

  S. J. v. Gool and B. Steinberg. "Pro-aperiodic monoids via saturated models". In: *STACS 2017*. Vol. 66. LIPIcs. https://arxiv.org/abs/1609.07736. 2017, 39:1–39:14

- A more category-theoretic view of formal language theory: (see also part II)

  M. Gehrke, D. Petrisan, and L. Reggio. "Quantifiers on languages and codensity monads". In: *TACL 2017*. https://arxiv.org/abs/1702.08841. 2017

Outline

1. Part I: Formal Languages, Automata, and Algebra
2. Part II: Duality and Varieties of Monoids
3. Part III: Profiniteness, Pointlikes, and the Future

Recap from Part I

- Formal Σ-languages are subsets of Σ^*, the set of finite words over a finite alphabet Σ.
- Finite-state automata (deterministic or not) describe the *regular* languages.
- Monadic second order logic also describes exactly the *regular* languages.
- First order logic describes a (strictly) smaller class of languages.
- The regular languages form a Boolean algebra with quotient operators.
- Every regular language L defines a finite closed Boolean subalgebra $B(L)$.
- Monoids are also somehow important (but why?)

Monoids

Examples

- The set Σ^*, with multiplication $u \cdot v := uv$.
- For any set P, the set of functions from P to itself, $(P \to P)$, with multiplication $f \cdot g := f \circ g$.
- In particular, an NFA $\mathcal{A} = (Q, \Sigma, \delta)$ gives, for every $a \in \Sigma$, a function $\Diamond_a$ in $(\mathcal{P}(Q) \to \mathcal{P}(Q))$, defined by

$$\Diamond_a(R) := \{q \mid q \xrightarrow{a} q' \text{ for some } q' \in R\}.$$

Exercises

1. Show that Σ^* is a monoid.
2. Show that $(P \to P)$ is a monoid.
3. Show that Σ^* is the *free* monoid on Σ, i.e., that for any monoid M and any function $f \colon \Sigma \to M$, there is a unique homomorphism $\bar{f} \colon \Sigma^* \to M$ extending f.
4. Applying (3) to the function $\Diamond \colon \Sigma \to (\mathcal{P}(Q) \to \mathcal{P}(Q))$, give an explicit description of the function $\bar{\Diamond} \colon \Sigma^* \to (\mathcal{P}(Q) \to \mathcal{P}(Q))$.
5. (*) Show that $\mathcal{A}$ with initial states I and final states F accepts a word $w \in \Sigma^*$ if, and only if, $I \cap \bar{\Diamond}_w(F) \neq \emptyset$.

Regular languages and monoids

Proposition

A Σ-language L is regular if, and only if, there exists a homomorphism $\eta \colon \Sigma^* \to M$, with M a finite monoid, such that $L = \eta^{-1}(R)$ for some $R \subseteq M$.

Proof ingredients.

- The exercises on the previous slide show how to build a monoid homomorphism from an NFA.
- For the converse, notice that a homomorphism from Σ^* to a monoid 'is' a (deterministic) automaton. □

Regular languages are:

- the languages recognized by finite non-deterministic automata.
- the languages recognized by finite deterministic automata.
- the languages definable in monadic second order logic.
- the inverse images of homomorphisms from the free monoid to a finite monoid.
- the unions of classes under finite index congruences on a free monoid.

Today, we will see how these characterizations are connected to each other through Stone duality.

Outline Part II

1. Finite Duality and Regular Languages
   - Boolean algebras
   - Finite Stone duality
   - Duality for regular languages

2. Full Duality and Varieties
   - First-order logic and aperiodic monoids
   - Full Stone duality

Stone duality

"In January last year I gave a course at the Indian Winter School in Logic and went on an excursion to Varanasi and Sarnath, the birthplace of Buddhism. Upon entering the amazing Archaeological Museum at Sarnath, our guide opened with: *'Duality underlies the world.'* This is the kind of sweeping statement that every mathematician, at least secretly, would like to believe about their particular focus..."

M. Gehrke. *Duality*. Oratie (inaugural lecture) at Radboud University Nijmegen, 2009. URL: `http://repository.ubn.ru.nl/bitstream/handle/2066/83300/83300.pdf`

Stone duality

- Stone duality was introduced by mathematician M. H. Stone in the 1930's.
- In logic, it underpins the connection between syntax and semantics.
- The dual of a collection of formulas (syntax) is a space of possible worlds/states (semantics) interpreting the formulas, and vice versa.
- A key idea, and the meaning of the term 'duality' (= dual categorical equivalence), is that the direction of morphisms is reversed.
- More information = Less possible worlds.
- More possible worlds = Less information.
- Formulating duality theory precisely requires some algebra, and, for the non-finite case, topology.
- We will focus on the applications to regular languages.

1. Finite Duality and Regular Languages
   - Boolean algebras
   - Finite Stone duality
   - Duality for regular languages

Boolean algebras

- An (abstract) *Boolean algebra* is a tuple $(B, \vee, \neg, \perp)$, where
  - B is a set,
  - $\vee$ is a binary operation,
  - $\neg$ is a unary operation,
  - $\perp$ is an element of B,
  - for any classical tautology $\varphi(\bar{x}) \leftrightarrow \psi(\bar{x})$ and $\bar{b}$ in B, $\varphi(\bar{b}) = \psi(\bar{b})$ in B.
- For example, $a \vee b = b \vee a$, $\neg\neg a = a$, $a \vee \perp = a$, $\ldots$.
- The last condition can be replaced by a finite list of axioms.
- Boolean algebras are partially ordered: $a \leq b$ iff $a \vee b = b$.

Boolean algebras: examples

Examples

- For any set X, $(\mathcal{P}(X), \cup, ()^{\mathrm{c}}, \emptyset)$ is a Boolean algebra.
- The *Lindenbaum algebra* of classical propositional logic on a set of variables V is the *free* Boolean algebra on V.
- For any topological space X, the *clopen* ($=$ closed and open) subsets are a Boolean subalgebra of $\mathcal{P}(X)$.

Finite Stone duality: algebras

Proposition

Every finite Boolean algebra B is isomorphic to a Boolean algebra of the form $\mathcal{P}(X)$, for some set X.

Proof.

Take $X = \mathrm{At}(B)$, the set of atoms of B.
Identify $b \in B$ with the set, $\hat{b}$, of atoms below it. □

Example

If $V = \{p_1, \ldots, p_n\}$, then the Lindenbaum algebra of classical propositional logic on V is isomorphic to $\mathcal{P}(X)$, where $X = \{0,1\}^V$.
In words: a formula of CPL can be identified with the set of valuations in which it is true.
When V is infinite, the situation is more subtle!

Finite Stone duality: homomorphisms

Proposition

Every homomorphism between finite Boolean algebras $\mathcal{P}(Y) \to \mathcal{P}(X)$ is of the form f^{-1} for some function $f \colon X \to Y$.

- In particular, any finite subalgebra of $\mathcal{P}(X)$ has the form $q^{-1} \colon \mathcal{P}(Y) \hookrightarrow \mathcal{P}(X)$, where $q \colon X \twoheadrightarrow Y$ is a quotient of X.
- In other words, any finite subalgebra of $\mathcal{P}(X)$ is the collection of finite unions of equivalence classes of an equivalence relation on X.

Subalgebras and equivalence relations

Example

- The closed subalgebra generated by the Σ-language $L = \texttt{EVENLENGTH}$ is

$$B(L) = \{\emptyset, L, L^{\mathrm{c}}, \Sigma^*\} \hookrightarrow \mathrm{Reg}(\Sigma^*).$$

- The dual of this subalgebra is a quotient $q \colon \Sigma^* \to \mathrm{At}\ B(L)$.
- This quotient is given by the equivalence relation $w_1 \equiv_L w_2$ if, and only if, the length of w_1 and w_2 have the same parity.

Finite Stone duality: regular languages

- Let L be a regular Σ-language.
- Let $B(L)$ be the finite closed subalgebra of $\mathrm{Reg}(\Sigma^*)$ generated by L.
- Then $B(L)$ is the set of unions of equivalence classes under an equivalence relation $\equiv_L$ on Σ^*, which can be defined by

$$w_1 \equiv_L w_2 \iff \text{for all } u, v \in \Sigma^*, uw_1v \in L \text{ iff } uw_2v \in L.$$

- A language $L \subseteq \Sigma^*$ is regular if, and only if, $\equiv_L$ has finite index.

Duality and regular languages

- $B(L)$ is a *closed* subalgebra of $\mathrm{Reg}(\Sigma^*)$.
- It follows that the dual $M(L) = \Sigma^*/{\equiv_L}$ of $B(L)$ is a monoid.
- The monoid $M(L)$ is the *syntactic monoid* of L.
- The homomorphism $q \colon \Sigma^* \to M(L)$ *recognizes* L: $L = q^{-1}(R)$ where $R = q(L)$.
- Moreover, $M(L)$ is the *minimum* such monoid quotient of Σ^*: if $q' \colon \Sigma^* \to M'$ recognizes L, then there exists $f \colon M' \to M(L)$ such that $fq' = q$.

Syntactic monoid: Example

Example

Let $\Sigma = \{0, 1\}$ and $L = \texttt{EVENLENGTH}$.

For $w_1, w_2 \in \Sigma^*$, $w_1 \equiv_L w_2$ iff the length of w_1 and of w_2 have the same parity.

Therefore, $M(L) \cong \mathbb{Z}_2$, the two-element group.

The quotient $q \colon \Sigma^* \to M(L)$ is defined by

$q(w) :=$ parity of the length of w.

Notice that $q(w_1 w_2) = q(w_1) \oplus q(w_2)$, i.e., q is a homomorphism.

Exercises

1. Find the syntactic monoid quotient $\Sigma^* \to M(L)$ when $L = \texttt{EVENONES}$.
2. Find the syntactic monoid quotient $\Sigma^* \to M(L)$ when $L = \texttt{BUY}$.
3. (*) Find the syntactic monoid quotient $\Sigma^* \to M(L)$ when $L = \texttt{PW}$.
4. Conclude from the solutions to (1) – (3) what the closed subalgebras, $B(L)$, generated by L are.
5. Use $\equiv_L$ to show that L is not regular when $L = \texttt{NON1}$.

2 Full Duality and Varieties

- First-order logic and aperiodic monoids
- Full Stone duality

FO and aperiodics

- In Part I, we asked: what is the subalgebra $\text{FO}(\Sigma^*)$ of $\mathrm{Reg}(\Sigma^*)$?
- We now know that any regular language L has a finite syntactic monoid $M(L)$.
- A monoid M is aperiodic if it contains no non-trivial subgroups.
- For finite monoids, it is equivalent to say:
  the equation $x^n = x^{n+1}$ holds in M for some n.
- It is also equivalent to say: $x^\omega = x^\omega x$,
  where x^ω is the idempotent power of x.

Theorem (Schützenberger, 1960s)

A language L is first-order definable if, and only if, the syntactic monoid $M(L)$ is finite and aperiodic.

An algorithm for deciding if a regular language is FO-definable.

Example of Schützenberger's Theorem

Example

- The syntactic monoid of `EVENLENGTH` is $\mathbb{Z}_2$.
- This contains (in fact, is) a group.
- By Schützenberger's theorem, `EVENLENGTH` is not first order definable.

Exercise

- Using the results from the previous exercise, determine which of the syntactic monoids for `EVENONES`, `BUY`, and `PW` are aperiodic.
- Conclude which of these languages are first order definable.

Varieties of monoids and languages

- A class of finite monoids **V** is a (pseudo)*variety* if it is closed under homomorphic images (H), submonoids (S) and finite products ($\mathrm{P}^{\mathrm{fin}}$).
- For a variety of monoids **V**, define $\mathcal{V}(\Sigma^*)$ to be the class of Σ-languages L such that $M(L) \in \mathbf{V}$.
- Then $\{\mathcal{V}(\Sigma^*)\}_\Sigma$ is a *variety of regular languages*: a collection of Boolean subalgebras of $\mathrm{Reg}(\Sigma^*)$ which is closed under inverse images of homomorphisms $\Sigma_1^* \to \Sigma_2^*$.

Theorem (Eilenberg)

The map $\mathbf{V} \mapsto \mathcal{V}$ is an order-bijection between varieties of finite monoids and varieties of regular languages.

Equations?

- Birkhoff's theorem: varieties of (arbitrary) algebras can be defined by (finite) equations.
- What about (pseudo)varieties of *finite* algebras?
- We need *pro*finite equations.
- To explain what these are, and why we need them: full Stone duality.

Stone duality: general case

Proposition

Every Boolean algebra B can be embedded into a Boolean algebra of the form $\mathcal{P}(X)$, and there is a unique such embedding for which the topology generated by the sets in the image of B is compact and Hausdorff (and zero-dimensional).

‘Construction’ of the embedding.

Take X to be the set of ultrafilters of B.
Identify $b \in B$ with the set, $\hat{b}$, of ultrafilters containing it. □

- A *Boolean space* is a compact Hausdorff zero-dimensional space.
- Equivalently, a *Boolean space* is a profinite object in the category of topological spaces.

Stone duality: example

Example

The dual space of the Lindenbaum algebra of CPL on a countable set $V = \{p_1, p_2, p_3, \dots\}$ is the *Cantor space* $\{0,1\}^V$.

Exercises

1. What is the dual space of the Boolean algebra of finite subsets of the natural numbers and their complements?
2. Use what you know about classical propositional logic to prove that the Lindenbaum algebra of CPL on a countable set $V = \{p_1, p_2, p_3, \dots\}$ can be embedded into $\mathcal{P}(\{0,1\}^V)$.
3. (*) Show that the topology generated by the image of the embedding in (2) is compact and Hausdorff.
4. (*) Show that the topology generated by the image of the embedding in (2) coincides with the topology of the Cantor space.

Duality: categorical level

- As in the finite case, all homomorphisms between Boolean algebras are of the form f^{-1}, for f a *continuous* function between the dual spaces.
- The *categories* of Boolean algebras and Boolean spaces are dually equivalent.

Algebras	dual to	Spaces
subalgebras	$\leftrightarrow$	quotient objects
quotient algebras	$\leftrightarrow$	subobjects
homomorphisms	$\leftrightarrow$	continuous functions
algebraic operations	$\leftrightarrow$	co-algebraic operations
unions (directed colimits)	$\leftrightarrow$	projective limits

Stone duality: summary

- Finite Boolean algebras are power sets.
- Boolean algebras are subalgebras of power sets.
- Boolean algebra homomorphisms are inverse images.
- Boolean algebras are algebras of clopen sets of a compact Hausdorff topological space, called the dual space.
- Subalgebras of the Boolean algebra correspond to quotient spaces of the dual space.
- Quotients of the Boolean algebra correspond to closed subspaces of the dual space.

References for Part II

- Basics on duality theory for Boolean algebras: Chapter 11 in

  B. A. Davey and H. A. Priestley. *Introduction to Lattices and Order*. 2nd. Cambridge University Press, May 6, 2002

- The duality-theoretic view on varieties:

  M. Gehrke. "Stone duality, topological algebra, and recognition". In: *J. Pure Appl. Algebra* 220.7 (2016), pp. 2711–2747

- A proof of Schützenberger's Theorem: Chapter VI in

  H. Straubing. *Finite automata, formal logic, and circuit complexity*. Progress in Theoretical Computer Science. Birkhäuser Boston Inc., Boston, 1994

- Our recent work on applications of model theory:

  S. J. v. Gool and B. Steinberg. "Pro-aperiodic monoids via saturated models". In: *STACS 2017*. Vol. 66. LIPIcs. https://arxiv.org/abs/1609.07736. 2017, 39:1–39:14

Outline

1. Part I: Formal Languages, Automata, and Algebra
2. Part II: Duality and Varieties of Monoids
3. Part III: Profiniteness, Pointlikes, and the Future

Recap

- Any regular language L comes with a finite closed Boolean subalgebra, $B(L)$, of $\mathrm{Reg}(\Sigma^*)$ and a finite monoid quotient, $M(L)$, of Σ^*.
- The syntactic monoid $M(L)$ is the Stone dual ($=$ atom set) of the closed Boolean subalgebra $B(L)$.
- Properties of the syntactic monoid reflect properties of the language.
- The syntactic monoid $M(L)$ is aperiodic iff L is first-order definable.
- In general, varieties (HSP^{fin}-classes) of monoids correspond to varieties of languages.
- Full (non-finite) Stone duality can be used to explain and analyze this correspondence but how?

Outline Part III

1. Full Duality and Varieties
   - Varieties
   - Full Stone duality
   - Intuitionistic intermezzo
   - Profinite equations
2. Aperiodic pointlikes
   - Separation problem
   - Pointlike sets
   - Henckell's Theorem
3. The Future

1. Full Duality and Varieties
    - Varieties
    - Full Stone duality
    - Intuitionistic intermezzo
    - Profinite equations

Varieties of monoids and languages

- A class of finite monoids $\mathbf{V}$ is a (pseudo)*variety* if it is closed under homomorphic images (H), submonoids (S) and finite products ($\mathsf{P}^{\mathrm{fin}}$).
- For a variety of monoids $\mathbf{V}$, define $\mathcal{V}(\Sigma^*)$ to be the class of Σ-languages L such that $M(L) \in \mathbf{V}$.
- Then $\{\mathcal{V}(\Sigma^*)\}_\Sigma$ is a *variety of regular languages*: a collection of Boolean subalgebras of $\mathrm{Reg}(\Sigma^*)$ which is closed under inverse images of homomorphisms $\Sigma_1^* \to \Sigma_2^*$.

Theorem (Eilenberg)

The map $\mathbf{V} \mapsto \mathcal{V}$ is an order-bijection between varieties of finite monoids and varieties of regular languages.

Equations?

- Birkhoff's theorem: varieties of (arbitrary) algebras can be defined by (finite) equations.
- What about (pseudo)varieties of *finite* algebras?
- We need *pro*finite equations.
- To explain what these are, and why we need them: full Stone duality.

Stone duality: general case

Proposition

Every Boolean algebra B can be embedded into a Boolean algebra of the form $\mathcal{P}(X)$, and there is a unique such embedding for which the topology generated by the sets in the image of B is compact and Hausdorff (and zero-dimensional).

'Construction' of the embedding.

Take X to be the set of ultrafilters of B.
Identify $b \in B$ with the set, $\hat{b}$, of ultrafilters containing it. □

- A *Boolean space* is a compact Hausdorff zero-dimensional space.
- Equivalently, a *Boolean space* is a profinite object in the category of topological spaces.

Stone duality: example

Example

The dual space of the Lindenbaum algebra of CPL on a countable set $V = \{p_1, p_2, p_3, \dots\}$ is the *Cantor space* $\{0,1\}^V$.

Exercises

1. What is the dual space of the Boolean algebra of finite subsets of the natural numbers and their complements?
2. Use what you know about classical propositional logic to prove that the Lindenbaum algebra of CPL on a countable set $V = \{p_1, p_2, p_3, \dots\}$ can be embedded into $\mathcal{P}(\{0,1\}^V)$.
3. (*) Show that the topology generated by the image of the embedding in (2) is compact and Hausdorff.
4. (*) Show that the topology generated by the image of the embedding in (2) coincides with the topology of the Cantor space.

Duality: categorical level

- As in the finite case, all homomorphisms between Boolean algebras are of the form f^{-1}, for f a *continuous* function between the dual spaces.
- The *categories* of Boolean algebras and Boolean spaces are *dually equivalent*.

Algebras	dual to	Spaces
subalgebras	$\leftrightarrow$	quotient objects
quotient algebras	$\leftrightarrow$	subobjects
homomorphisms	$\leftrightarrow$	continuous functions
algebraic operations	$\leftrightarrow$	co-algebraic operations
unions (directed colimits)	$\leftrightarrow$	projective limits

Stone duality: summary

- Finite Boolean algebras are power sets.
- Boolean algebras are subalgebras of power sets.
- Boolean algebra homomorphisms are inverse images.
- Boolean algebras are algebras of clopen sets of a compact Hausdorff topological space, called the dual space.
- Subalgebras of the Boolean algebra correspond to quotient spaces of the dual space.
- Quotients of the Boolean algebra correspond to closed subspaces of the dual space.

Intuitionistic Intermezzo

An open mapping theorem for Esakia spaces

- Stone duality generalizes to *Heyting algebras*, the structures for *intuitionistic* propositional logic analogous to Boolean algebras.
- The Boolean space is equipped with a *partial order* (= the Kripke accessibility relation in semantics for intuitionistic logic).
- The spaces dual to Heyting algebras were characterized by L. Esakia and are now called *Esakia spaces*.
- Heyting algebra homomorphisms also require special attention: their duals are *continuous p-morphisms*.
- Esakia duality is useful, for example, for proving *interpolation* properties of intermediate logics.
- In recent joint work with L. Reggio, we prove an *open mapping theorem* for Esakia spaces dual to finitely presented Heyting algebras.
- Our result in particular implies Pitts' Uniform Interpolation Thoerem for IPC.

Another hint of duality for interpolation?

- The use of duality for analyzing quantifiers is not limited to the context of regular languages.
- A classical example is Rasiowa & Sikorski's proof of the completeness of classical predicate logic via Stone duality and the Baire category theorem.
- A recent example (I claim) is the counterexample to interpolation for constant domain intuitionistic predicate logic (Mints, Olkhovikov, Urquhart JSL 2013).
- In the latter, and in Olkhovikov's work on van-Benthem-style characterizations (2012-2015), the use of duality is not (yet) explicit.

End of Intuitionistic Intermezzo

Stone duality: crucial example for language varieties

Example

The dual space of the Boolean algebra $\mathrm{Reg}(\Sigma^*)$ of regular Σ-languages is the projective limit of the diagram $(q\colon \Sigma^* \to M)$ of *finite* quotients of Σ^*. This is the space underlying the free profinite monoid, $\widehat{\Sigma^*}$, on Σ.
Thus, the free profinite monoid over Σ is the 'canonical Kripke model' for MSO on finite words (since the Lindenbaum algebra is $\mathrm{Reg}(\Sigma^*)$).

On the free profinite monoid

"In the mid-1970s when I was at Oxford, it occurred to me that, using regular events in the free monoid on a finite alphabet as neighborhoods, one could make a completion to 'infinite words'. I even suggested to one of my students to consider this idea for a thesis. Neither he nor I could make much progress in analyzing this algebra or applying this idea, however, and he went on to write a different thesis (fortunately)."
[D. Scott, via e-mail, Nov. 9, 2016]

Describing varieties: profinite equations

- Let **V** be a variety of finite monoids.
- Let $\mathcal{V}$ be the corresponding variety of regular languages.
- For every alphabet Σ, $\mathcal{V}(\Sigma^*)$ is a closed subalgebra of $\mathrm{Reg}(\Sigma^*)$.
- The dual of this closed subalgebra is a continuous monoid quotient $\widehat{\Sigma^*} \to \widehat{F}_{\mathbf{V}}(\Sigma)$, the free pro-**V**-monoid on Σ.
- The fact that $\mathcal{V}$ is closed under inverse images of homomorphisms means that the quotients $\widehat{\Sigma^*} \to \widehat{F}_{\mathbf{V}}(\Sigma)$ are substitution-invariant.
- Thus, varieties can be described by profinite equations.

Profinite equations: example

Example

The free pro-aperiodic monoid, $\widehat{F}_{\mathbf{A}}(\Sigma^*)$, is the quotient of $\widehat{\Sigma^*}$ by the equivalence relation defined by the substitution-invariant equation $x^\omega = x^\omega x$.
Here, $()^\omega \colon \widehat{\Sigma^*} \to \widehat{\Sigma^*}$ is the operation which sends any x to the idempotent x^ω in $\overline{\{x^n \mid n \geq 1\}}$.

Duality beyond the profinite

- Let $\mathcal{V}$ be any closed subalgebra of $\mathcal{P}(\Sigma^*)$.
- It corresponds again to a topological quotient, but now of $\beta\Sigma^*$, the Stone-Cech compactification of the discrete free monoid.
- The story for the monoid operation is a bit more complicated (internal monoid action).
- This idea, combined with methods from circuit complexity theory, leads to 'ultrafilter equations' for characterizing classes of non-regular languages.
- M. Gehrke, A. Krebs, and J.-É. Pin. "From ultrafilters on words to the expressive power of a fragment of logic". In: *DCFS 2014*. Vol. 8614. Lect. Notes Comput. Sci. Springer, 2014, pp. 138–149
- M. Gehrke, D. Petrisan, and L. Reggio. "Quantifiers on languages and codensity monads". In: *TACL 2017*. https://arxiv.org/abs/1702.08841. 2017

2 Aperiodic pointlikes

- Separation problem
- Pointlike sets
- Henckell's Theorem

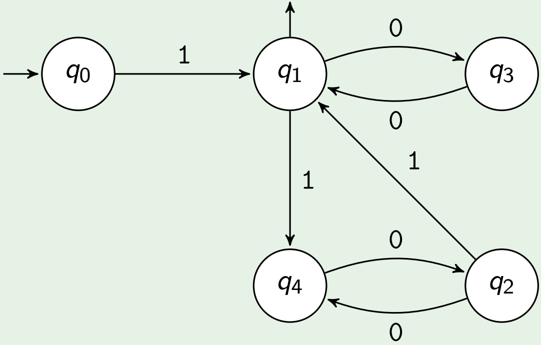
Separation problem: language version

- From here on, we work with semigroups, and ‘Σ-language’ means subset of Σ^+.
- Let $\mathbf{V}$ be a variety of finite semigroups with corresponding variety of languages $\mathcal{V}$.
- Separation Problem: Given two disjoint regular Σ-languages L_1, L_2, is it possible to find a language, K, in $\mathcal{V}(\Sigma)$ which *separates* L_1 from L_2?
- Here, K separates L_1 from L_2 if $L_1 \subseteq K$ and $L_2 \cap K = \emptyset$.
- If φ_1 and φ_2 are MSO sentences defining L_1 and L_2, respectively, then disjointness means $\varphi_1 \vdash \neg\varphi_2$.
- The *logic formulation* of the separation problem is: does there exist ψ such that $\varphi_1 \vdash \psi \vdash \neg\varphi_2$, with the language $K = L_\psi$ in $\mathcal{V}(\Sigma)$.
- In general, this problem can fail to be decidable, even when membership in $\mathcal{V}$ is decidable.

Example of non-separable languages

Example (Place & Zeitoun 2016)

Let $\Sigma = \{0, 1\}$. Consider the automaton

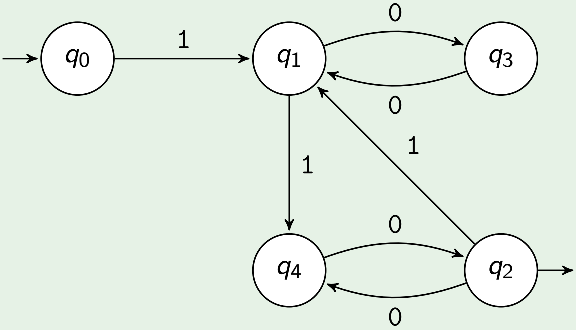


The language recognized with q_1 final is $L_1 = (1(00)^*10(00)^*)^*1(00)^*$.

Example of non-separable languages

Example (Place & Zeitoun 2016)

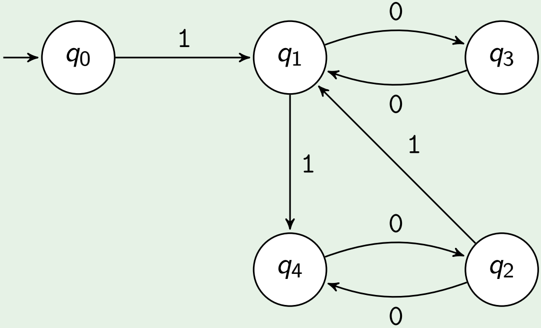
Let $\Sigma = \{0, 1\}$. Consider the automaton



The language recognized with q_1 final is $L_1 = (1(00)^*10(00)^*)^*1(00)^*$.
The language recognized with q_2 final is $L_2 = (1(00)^*10(00)^*)^+$.

Example (Place & Zeitoun 2016)

Let $\Sigma = \{0, 1\}$. Consider the automaton



The language recognized with q_1 final is $L_1 = (1(00)^*10(00)^*)^*1(00)^*$.

The language recognized with q_2 final is $L_2 = (1(00)^*10(00)^*)^+$.

The languages L_1 and L_2 are *disjoint*, but not FO-separable.

Exercise: Use first-order logic games to prove this.

Separation problem: semigroup version

- The separation problem can be formulated as a problem about semigroups.
- We may assume L_1 and L_2 are recognized by the same semigroup homomorphism $\eta\colon \Sigma^+ \to S$.
- So $L_1 = \eta^{-1}(R_1)$ and $L_2 = \eta^{-1}(R_2)$, with R_1 and R_2 disjoint.
- Is there a semigroup homomorphism $\theta\colon \Sigma^* \to T \in \mathbf{V}$, and $P \subseteq T$, such that $\eta^{-1}(R_1)$ is contained in $\theta^{-1}(P)$, and $\eta^{-1}(R_2)$ is disjoint from $\theta^{-1}(P)$?
- Fact. The answer is 'no' if, and only if, for every $r_1 \in R_1$ and $r_2 \in R_2$, the subset $\{r_1, r_2\}$ of S is pointlike.

Pointlike sets

- A relational morphism from a semigroup S to a semigroup T is a relation $\varphi \subseteq S \times T$ such that $s\varphi \cdot s'\varphi \subseteq ss'\varphi$ and $s\varphi \neq \emptyset$ for all $s, s' \in S$.
- Equivalently, it is a relation of the form $\beta\alpha^{-1}$, where $\alpha \colon U \twoheadrightarrow S$ and $\beta \colon U \to T$ are homomorphisms from a semigroup U.
- A subset $X \subseteq S$ is **V**-pointlike if, for every relational morphism $\varphi \colon S \to T$ such that $T \in \mathbf{V}$, there exists a point $x \in T$ such that $X \subseteq \varphi^{-1}(x)$.
- If we can compute the (two-element) **V**-pointlike sets of S, then we can decide the **V**-separation problem:
- Given $L_1 = \eta^{-1}(R_1)$, $L_2 = \eta^{-1}(R_2)$ for $\eta \colon \Sigma^* \to M$, check if $\{r_1, r_2\}$ is pointlike for all $r_1 \in R_1$, $r_2 \in R_2$. If so, L_1 and L_2 are non-separable.
- In particular, to decide FO-separation, we will compute the **A**-pointlike sets, where **A** is the variety of aperiodic semigroups.

The monad of **V**-pointlikes

- The collection of **V**-pointlike sets, $\mathrm{PL}_{\mathbf{V}}(S)$, of a semigroup S is a subset of the *power semigroup*, 2^S, of S.
- Elements of 2^S are subsets of S, i.e., as a set, $2^S = \mathcal{P}(S)$.
- Multiplication on 2^S is given by: $X \cdot Y = \{xy \mid x \in X, y \in Y\}$.

Fact

*The collection, $\mathrm{PL}_{\mathbf{V}}(S)$, of **V**-pointlike subsets of a finite semigroup S, is a downward closed subsemigroup of 2^S which contains all the singletons.*

Fact

*The union of a **V**-pointlike subset of the semigroup $\mathrm{PL}_{\mathbf{V}}(S)$ is **V**-pointlike. That is, $\bigcup\colon \mathrm{PL}_{\mathbf{V}}(\mathrm{PL}_{\mathbf{V}}(S)) \to \mathrm{PL}_{\mathbf{V}}(S)$ is well-defined.*

Generating aperiodic-pointlike sets

- For $X \in 2^S$, define $X^{\omega+*} = \bigcup_{n \geq 0} X^\omega X^n$.
- Fact. If X is **A**-pointlike, then so is $X^{\omega+*}$.
- Singletons are **A**-pointlike.
- Products of **A**-pointlike sets are **A**-pointlike.
- Subsets of **A**-pointlike sets are **A**-pointlike.

Theorem (Henckell)

For any finite semigroup S, the set of **A***-pointlikes of S is the smallest downward closed subsemigroup of 2^S which contains the singletons and is closed under the operation $X \mapsto X^{\omega+*}$.*

In particular, the **A**-pointlikes of any finite semigroup are computable.

Recent progress

- Henckell, Rhodes and Steinberg (2010) improved on Henckell's original proof and extended his methods to varieties of semigroups that avoid specific subgroups.
- Place and Zeitoun (2016) gave a logic proof of Henckell's Thm.
- Place and Zeitoun (2014-17) computed FO_2-pointlikes.
- Steinberg and I (2017) gave a semigroup proof of Henckell's Thm.
- To do so, we construct a 'merge decomposition' of homomorphisms.
- This is an algebraic version of 'quantifying over first and last occurrences'.
- In addition to a short elementary proof of Henckell's Theorem, we also give a short proof of the two-sided Krohn-Rhodes theorem.
- The latter, in a slogan, says:
  'semigroup theory = semilattice theory + group theory'.

References for Part III

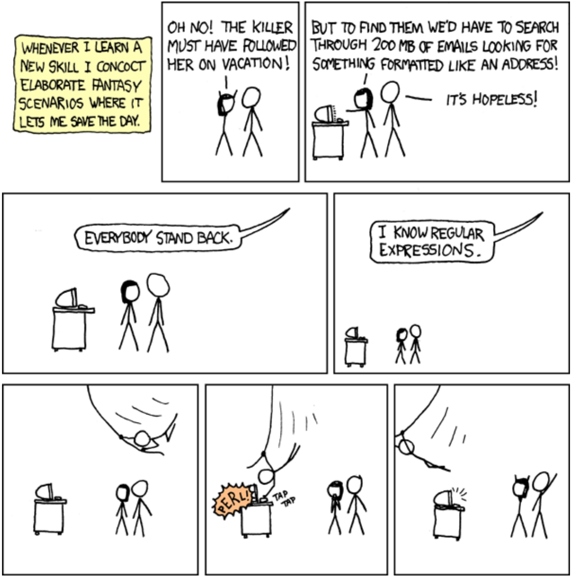
- Basics on duality theory for Boolean algebras: Chapter 11 in
  B. A. Davey and H. A. Priestley. *Introduction to Lattices and Order*. 2nd. Cambridge University Press, May 6, 2002
- The duality-theoretic view on varieties:
  M. Gehrke. “Stone duality, topological algebra, and recognition”. In: *J. Pure Appl. Algebra* 220.7 (2016), pp. 2711–2747
- On the relation between pointlikes and separation:
  J. Almeida. “Some algorithmic problems for pseudovarieties”. In: *Publ. Math. Debrecen* 54.1 (1999), pp. 531–552
- Our recent preprint on Henckell’s and Krohn-Rhodes theorems:
  S. J. v. Gool and B. Steinberg. “Merge decompositions, two-sided Krohn-Rhodes, and aperiodic pointlikes”. arXiv:1708.08118, submitted. Aug. 2017

3 The Future

Seven questions

- How far can the decidability of pointlikes be stretched?
- How far do duality-theoretic methods reach beyond the regular?
- How does our semigroup-theoretic work fit with the category/duality approach?
- Is there a topos-theoretic interpretation of 'logic on words'?
- What can be said about regular languages with model theory? (Partial answers in joint work with Ghilardi)
- Can the relationship with modal logic be made more tight?
- Is anything I've said relevant for (formal) linguistics?

გმადლობ


WHENEVER I LEARN A NEW SKILL I CONCOCT ELABORATE FANTASY SCENARIOS WHERE IT LETS ME SAVE THE DAY.
OH NO! THE KILLER MUST HAVE FOLLOWED HER ON VACATION!
BUT TO FIND THEM WE'D HAVE TO SEARCH THROUGH 200 MB OF EMAILS LOOKING FOR SOMETHING FORMATTED LIKE AN ADDRESS!
IT'S HOPELESS!
EVERYBODY STAND BACK.
I KNOW REGULAR EXPRESSIONS.
PERL!
TAP TAP

XKCD 208: Regular Expressions (`https://xkcd.com/208/`)