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

Recap from Part I

- Formal Σ-languages are subsets of Σ^*, the set of finite words over a finite alphabet Σ.

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.

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.

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.

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.

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 $B(L)$.

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

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 $u \cdot v := uv$.

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 $f \cdot g := f \circ g$.

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 $R \subseteq M$.

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:

Regular languages are:

- the languages recognized by finite non-deterministic automata.

Regular languages are:

- the languages recognized by finite non-deterministic automata.
- the languages recognized by finite deterministic automata.

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.

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.

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.

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.

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.

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.

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.

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.

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.

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.

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 $(B, \vee, \neg, \bot)$, where
  - B is a set,
  - $\vee$ is a binary operation,
  - $\neg$ is a unary operation,
  - $\bot$ 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.

Boolean algebras

- An (abstract) *Boolean algebra* is a tuple $(B, \vee, \neg, \bot)$, where
  - B is a set,
  - $\vee$ is a binary operation,
  - $\neg$ is a unary operation,
  - $\bot$ 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, $a \vee b = b \vee a$, $\neg\neg a = a$, $a \vee \bot = a$, $\ldots$.

Boolean algebras

- An (abstract) *Boolean algebra* is a tuple $(B, \vee, \neg, \bot)$, where
  - B is a set,
  - $\vee$ is a binary operation,
  - $\neg$ is a unary operation,
  - $\bot$ 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, $a \vee b = b \vee a$, $\neg\neg a = a$, $a \vee \bot = a$, $\ldots$.
- The last condition can be replaced by a finite list of axioms.

Boolean algebras

- An (abstract) *Boolean algebra* is a tuple $(B, \vee, \neg, \bot)$, where
  - B is a set,
  - $\vee$ is a binary operation,
  - $\neg$ is a unary operation,
  - $\bot$ 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, $a \vee b = b \vee a$, $\neg\neg a = a$, $a \vee \bot = a$, $\ldots$.
- 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, $(\mathcal{P}(X), \cup, ()^{c}, \emptyset)$ is a Boolean algebra.

Boolean algebras: examples

Examples

- For any set X, $(\mathcal{P}(X), \cup, ()^{c}, \emptyset)$ is a Boolean algebra.
- The *Lindenbaum algebra* of classical propositional logic on a set of variables V is the *free* Boolean algebra on V.

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.

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.

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.

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 $X = \{0, 1\}^V$.

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.

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* $f\colon X \to Y$*.*

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

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

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.

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.

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^*

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

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

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 $\mathrm{Reg}(\Sigma^*)$.

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.

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.

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 $R = q(L)$.

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 $L = \texttt{EVENLENGTH}$.

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

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.

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.

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

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.

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 $L = \texttt{PW}$.

Exercises

1. Find the syntactic monoid quotient $\Sigma^* \to M(L)$ when $L = \mathtt{EVENONES}$.
2. Find the syntactic monoid quotient $\Sigma^* \to M(L)$ when $L = \mathtt{BUY}$.
3. (*) Find the syntactic monoid quotient $\Sigma^* \to M(L)$ when $L = \mathtt{PW}$.
4. Conclude from the solutions to (1) – (3) what the closed subalgebras, $B(L)$, generated by L are.

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 $\mathsf{FO}(\Sigma^*)$ of $\mathrm{Reg}(\Sigma^*)$?

FO and aperiodics

- In Part I, we asked: what is the subalgebra $\mathsf{FO}(\Sigma^*)$ of $\mathrm{Reg}(\Sigma^*)$?
- We now know that any regular language L has a finite syntactic monoid $M(L)$.

FO and aperiodics

- In Part I, we asked: what is the subalgebra $\mathsf{FO}(\Sigma^*)$ 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.

FO and aperiodics

- In Part I, we asked: what is the subalgebra $\mathsf{FO}(\Sigma^*)$ 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.

FO and aperiodics

- In Part I, we asked: what is the subalgebra $\mathsf{FO}(\Sigma^*)$ 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.

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.

FO and aperiodics

- In Part I, we asked: what is the subalgebra $\mathsf{FO}(\Sigma^*)$ 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 $\mathbb{Z}_2$.

Example of Schützenberger's Theorem

Example

- The syntactic monoid of EVENLENGTH is $\mathbb{Z}_2$.
- This contains (in fact, is) a group.

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 ($\mathsf{P}^{\mathrm{fin}}$).

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 $M(L) \in \mathbf{V}$.

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 $\Sigma_1^* \to \Sigma_2^*$.

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 (P^{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.

Equations?

- Birkhoff's theorem: varieties of (arbitrary) algebras can be defined by (finite) equations.
- What about (pseudo)varieties of *finite* algebras?

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.

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 $\mathcal{P}(X)$

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).

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.

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.

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.

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.

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.

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

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

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

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

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

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.

Stone duality: summary

- Finite Boolean algebras are power sets.
- Boolean algebras are subalgebras of power sets.

Stone duality: summary

- Finite Boolean algebras are power sets.
- Boolean algebras are subalgebras of power sets.
- Boolean algebra homomorphisms are inverse images.

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.

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.

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

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

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

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