Open Universiteit

Institute for Logic, Language and Computation

Completeness and the FMP for KA, revisited

Tobias Kappé

OUrsi, March 28, 2023

Some context

- Program semantics sometimes obey laws of Kleene algebra (KA).
- What can we (not) prove using these laws?
- Can we automate proof search for pure KA equivalences?

Definition (Kleene algebra)

A *Kleene algebra* is a tuple $(K, +, \cdot, {}^*, 0, 1)$ where K is a set with $0, 1 \in K$. Also, $+$, $\cdot$ and * are respectively binary, binary and unary operators on K, satisfying

$$x + 0 = x \qquad x + x = x \qquad x + y = y + x \qquad x + (y + z) = (x + y) + z$$

$$x \cdot (y \cdot z) = (x \cdot y) \cdot z \qquad x \cdot (y + z) = x \cdot y + x \cdot z \qquad (x + y) \cdot z = x \cdot z + y \cdot z$$

$$x \cdot 1 = x = 1 \cdot x \qquad x \cdot 0 = 0 = 0 \cdot x \qquad 1 + x \cdot x^* = x^* \qquad x + y \cdot z \leq z \implies y^* \cdot x \leq z$$

Here, $x \leq y$ is a shorthand for $x + y = y$.

Kleene algebra

Languages

Fix a (finite) set of *letters* Σ.

Example (KA of languages)

The KA of *languages over* Σ is given by $(\mathcal{P}(\Sigma^*), \cup, \cdot, {}^*, \emptyset, \{\epsilon\})$, where

- Σ^* is the set of *words* with letters from Σ;
- $\mathcal{P}(\Sigma^*)$ is the set of sets of words (*languages*);
- $\cdot$ is pointwise concatenation, i.e., $L \cdot K = \{wx : w \in L, x \in K\}$;
- * is the Kleene star, i.e., $L^* = \{w_1 \cdots w_n : w_1, \ldots, w_n \in L\}$;
- ϵ is the empty word.

Fix a (not necessarily finite) set of *states* S.

Example (KA of relations)

The KA of *relations over* S is given by $(\mathcal{P}(S \times S), \cup, \circ, {}^*, \emptyset, \Delta)$, where

- $\mathcal{P}(S \times S)$ is the set of relations on S;
- $\circ$ is relational composition, i.e., $u(P \circ Q)w \iff \exists v.\ u\ P\ v \wedge v\ Q\ w$.
- * is the reflexive-transitive closure, i.e., $P^* = \Delta \cup P \cup P \circ P \cup \cdots$
- Δ is the identity relation.

Kleene algebra

Model theory

Definition

We write Exp for the set of *regular expressions*, generated by

$$e, f ::= 0 \mid 1 \mid \mathtt{a} \in \Sigma \mid e + f \mid e \cdot f \mid e^*$$

Definition

Given a KA $(K, +, \cdot, ^*, 0, 1)$ and $h : \Sigma \to K$, we define $\hat{h} : \mathrm{Exp} \to K$ by

$$\begin{aligned} \hat{h}(0) &= 0 & \hat{h}(e + f) &= \hat{h}(e) + \hat{h}(f) \\ \hat{h}(1) &= 1 & \hat{h}(e \cdot f) &= \hat{h}(e) \cdot \hat{h}(f) \\ \hat{h}(\mathtt{a}) &= h(\mathtt{a}) & \hat{h}(e^*) &= \hat{h}(e)^* \end{aligned}$$

Let $e, f \in \mathrm{Exp}$; we write $K \models e = f$ when $\hat{h}(e) = \hat{h}(f)$ for all h.

Kleene algebra

Model theory

Let $e, f \in \mathrm{Exp}$. We write ...

- $\vdash e = f$ when $e = f$ follows from the axioms of KA.
- $\models e = f$ when $K \models e = f$ for every KA K.
- $\mathfrak{R} \models e = f$ when $\mathcal{P}(S \times S) \models e = f$ for all S.
- $\mathfrak{F} \models e = f$ when $K \models e = f$ holds in every finite KA K.

Kleene algebra

Model theory

$$
\begin{tikzcd}
\mathfrak{F} \models e = f \arrow[r, Rightarrow, "\text{(Palka 2005)}", "\text{FMP}"'] & \vdash e = f \arrow[d, Rightarrow] & \mathcal{P}(\Sigma^*) \models e = f \arrow[l, Rightarrow, "\text{Completeness}", "\text{(Kozen 1994)}"'] \arrow[d, Leftrightarrow, "\text{(Pratt 1980)}"] \\
& \models e = f \arrow[ul, Rightarrow] \arrow[ur, Rightarrow] & \mathfrak{R} \models e = f
\end{tikzcd}
$$

This paper

Palka's proof of the FMP relies on Kozen's completeness theorem.

> *. . . an independent proof of [the finite model property] would provide a quite different proof of the Kozen completeness theorem, based on purely logical tools. We defer this task to further research.* (Palka 2005)

This paper gives that proof — with many ideas inspired by Palka.

Roadmap

Forward direction is easy

$$\vdash e = f \implies \models e = f \implies \mathfrak{F} \models e = f$$

The converse is harder! Given e, f, we do the following:

1. Turn expressions e, f into a finite automaton A
2. Turn the finite automaton A into a finite monoid M
3. Turn the finite monoid M into a finite KA K

Expressions to automata

Definition

An automaton is a tuple $(Q, \to, I, F)$ where

- Q is a finite set of *states*; and
- $\to \subseteq Q \times \Sigma \times Q$ is the *transition relation*; and
- $I \subseteq Q$ is the set of *initial states*
- $F \subseteq Q$ is the set of *accepting states*

We write $q \xrightarrow{\mathtt{a}} q'$ when $(q, \mathrm{a}, q') \in \to$.

Expressions to automata

Definition

Let $(Q, \to, F)$ be an automaton. A *solution* is a function $s : Q \to \mathsf{Exp}$ such that

$$\vdash F(q) + \sum_{q \xrightarrow{\mathtt{a}} q'} \mathtt{a} \cdot s(q') \leq s(q)$$

Here, $F(q) = 1$ when $q \in F$ and $F(q) = 0$ otherwise.

Theorem (Kleene 1956; see also Conway 1971)

Every automaton admits a unique least solution.

Expressions to automata

When A is an automaton, we write $A(q)$ for its least solution at q.

Lemma

For every e, we can construct an automaton $A_e = (Q_e, \to_e, I_e, F_e)$ such that

$$\vdash e = \sum_{q \in I_e} A_e(q)$$

NB: Like (Kozen 2001) and (Jacobs 2006) except with nondeterminism!

Automata to monoids

Definition

A *monoid* is a triple $(M, \cdot, 1)$ where M is a set, $\cdot$ is a binary operator, and $1 \in M$ s.t.

$$x \cdot (y \cdot z) = (x \cdot y) \cdot z \qquad\qquad x \cdot 1 = x = 1 \cdot x$$

Example (Free monoid)

$(\Sigma^*, \cdot, \epsilon)$ is a monoid.

Example (Transition monoid; McNaughton and Papert 1968)

If $A = (Q, \rightarrow, I, F)$ is an automaton, then $(M_A, \circ, \Delta)$ is a monoid, where

$$M_A = \{\xrightarrow{\mathtt{a}_1} \circ \cdots \circ \xrightarrow{\mathtt{a}_n} : \mathtt{a}_1, \ldots, \mathtt{a}_n \in \Sigma\}$$

Automata to monoids

Definition (Transition automaton)

Let $A = (Q, \to, I, F)$ be an automaton and $R \in M_A$.

We write $A[R]$ for the *transition automaton* $(M_A, \to_\circ, \Delta, \{R\})$ where

$$P \xrightarrow{\mathtt{a}}_\circ Q \iff P \circ \xrightarrow{\mathtt{a}} = Q$$

Lemma

Let $A = (Q, \to, I, F)$ be an automaton; it holds that

$$\vdash A(q) = \sum_{qRq_f \in F} A[R](\Delta)$$

Monoids to Kleene algebras

Lemma (Palka 2005)

Let $(M, \cdot, 1)$ *be a monoid. Now* $(\mathcal{P}(M), \cup, \otimes, {}^{\circledast}, \emptyset, \{1\})$ *is a KA, where*

$$T \otimes U = \{t \cdot u : t \in T \wedge u \in U\} \qquad T^{\circledast} = \{t_1 \cdots t_n : t_1, \ldots, t_n \in T\}$$

Lemma

Let A *be an automaton, and let* $h : \Sigma \to \mathcal{P}(M_A)$ *where* $h(\mathtt{a}) = \{\xrightarrow{\mathtt{a}}\}$*. Now*

$$R \in \hat{h}(A(q)) \iff q \; R \; q_f \in F$$

Putting it all together

In the sequel, fix $e, f \in \mathsf{Exp}$, and:

- Let $A_{e,f} = (Q_{e,f}, \rightarrow_{e,f}, I_{e,f}, F_{e,f})$ be the disjoint union of A_e and A_f.
- Let $M_{e,f} = (M_{A_{e,f}}, \circ, \Delta)$ be the monoid of $A_{e,f}$.

Lemma (Normal form)

Let $e, f \in \mathsf{Exp}$ and $h : \Sigma \to \mathcal{P}(M_{e,f})$ be given by $h(\mathtt{a}) = \{\xrightarrow{\mathtt{a}}_{e,f}\}$. The following hold:

$$\vdash e = \sum_{R \in \hat{h}(e)} A_{e,f}[R](\Delta) \qquad\qquad \vdash f = \sum_{R \in \hat{h}(f)} A_{e,f}[R](\Delta)$$

Putting it all together

Finite model property

Theorem (Finite model property)

If $\mathfrak{F} \models e = f$ *then* $\vdash e = f$*.*

Proof.

$\mathcal{P}(M_{e,f})$ is a finite KA; hence $\mathcal{P}(M_{e,f}) \models e = f$, i.e., $\hat{h}(e) = \hat{h}(f)$. But then:

$$\vdash e = \sum_{R \in \hat{h}(e)} A_{e,f}[R](\Delta) = \sum_{R \in \hat{h}(f)} A_{e,f}[R](\Delta) = f \qquad \square$$

Putting it all together

Completeness

Theorem (Completeness)

If $\mathcal{P}(\Sigma^*) \models e = f$ *then* $\vdash e = f$.

Proof.

Let $L : \Sigma \to \mathcal{P}(\Sigma^*)$ be given by $L(\mathtt{a}) = \{\mathtt{a}\}$.

We can show that $\hat{h}(e) = \{\xrightarrow{\mathtt{a}_1}_{e,f} \circ \cdots \circ \xrightarrow{\mathtt{a}_n}_{e,f} : \mathtt{a}_1 \cdots \mathtt{a}_n \in \hat{L}(e)\}$, and similarly for f.

If $\mathcal{P}(\Sigma^*) \models e = f$, then $\hat{L}(e) = \hat{L}(e)$, so $\hat{h}(e) = \hat{h}(f)$. The rest proceeds as before. $\square$

Coq formalization

- All results formalized in the Coq theorem prover.
- Trusted base:
  - Calculus of Inductive Constructions.
  - Streicher's *axiom K*.
  - Dependent functional extensionality.
- Some concepts are encoded differently; ideas remain the same.

Open questions

- Can we apply these ideas to *guarded Kleene algebra with tests*?
- Does KA have a *finite relational model property*?
- Do these techniques extend to *KA with hypotheses*?

References I

- Conway, John Horton (1971). *Regular Algebra and Finite Machines*. Chapman and Hall, Ltd., London.
- Jacobs, Bart (2006). "A Bialgebraic Review of Deterministic Automata, Regular Expressions and Languages". In: *Algebra, Meaning, and Computation, Essays Dedicated to Joseph A. Goguen on the Occasion of His 65th Birthday*, pp. 375–404. DOI: 10.1007/11780274_20.
- Kleene, Stephen C. (1956). "Representation of Events in Nerve Nets and Finite Automata". In: *Automata Studies*, pp. 3–41.
- Kozen, Dexter (1994). "A Completeness Theorem for Kleene Algebras and the Algebra of Regular Events". In: *Inf. Comput.* 110.2, pp. 366–390. DOI: 10.1006/inco.1994.1037.
- — (2001). "Myhill-Nerode Relations on Automatic Systems and the Completeness of Kleene Algebra". In: *STACS*, pp. 27–38. DOI: 10.1007/3-540-44693-1_3.

References II

McNaughton, Robert and Seymour Papert (1968). “The syntactic monoid of a regular event”. In: *Algebraic Theory of Machines, Languages, and Semigroups*, pp. 297–312.

Palka, Ewa (2005). “On Finite Model Property of the Equational Theory of Kleene Algebras”. In: *Fundam. Informaticae* 68.3, pp. 221–230. URL: http://content.iospress.com/articles/fundamenta-informaticae/fi68-3-02.

Pratt, Vaughan R. (1980). “Dynamic Algebras and the Nature of Induction”. In: *STOC*, pp. 22–28. DOI: 10.1145/800141.804649.