Open Universiteit

Institute for Logic, Language and Computation

Completeness and the FMP for KA, revisited

Tobias Kappé

(i)Po(m)set Project Online Seminar, June 16, 2023

Some context

- Laws of Kleene algebra apply to many programming language semantics.
- This means we can use KA to reason about program semantics.
- What can we (not) prove using these laws?
- When is something not true *only by the laws of KA*?

Definition (Kleene algebra)

A *Kleene algebra* is a tuple $(K, +, \cdot, ^*, 0, 1)$ where for all $x, y, z \in K$, we have:

$$x + x = x \qquad 1 + x \cdot x^* = x^* \qquad 1 + x^* \cdot x = x^* \qquad \frac{x + y \cdot z \leq z}{y^* \cdot x \leq z} \qquad \frac{x + y \cdot z \leq y}{x \cdot z^* \leq y}$$

in addition to the “usual” laws for $+$ and $\cdot$ — associativity, distributivity, etc.

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

Kleene algebra

Languages

Fix a (finite) set of *letters* Σ, and write Σ^* for the set of words over Σ.

Example (KA of languages)

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

- $\mathcal{P}(\Sigma^*)$ 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.
- * is the reflexive-transitive closure.
- Δ is the identity relation.

Kleene algebra

Reasoning example

Claim

In every KA K and for all $u, v \in K$, it holds that $(u \cdot v)^* \cdot u \leq u \cdot (v \cdot u)^*$.

Proof. First, let's recall the fixpoint rule:

$$\frac{x + y \cdot z \leq z}{y^* \cdot x \leq z}$$

It suffices to prove that $u + u \cdot v \cdot u \cdot (v \cdot u)^* \leq u \cdot (v \cdot u)^*$; we derive:

$$u + u \cdot v \cdot u \cdot (v \cdot u)^* = u \cdot (1 + v \cdot u \cdot (v \cdot u)^*) = u \cdot (v \cdot u)^* \qquad \square$$

Kleene algebra

Expressions

Definition

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

Examples

- We showed just now that $K \models (\mathtt{a} \cdot \mathtt{b})^* \cdot \mathtt{a} \leq \mathtt{a} \cdot (\mathtt{b} \cdot \mathtt{a})^*$ for all KAs K.
- $\mathcal{P}(\Sigma^*) \models e = f$ when e and f denote the same regular language.
- $\mathcal{P}(S \times S) \models (\mathtt{a} + 1)^* = \mathtt{a}^*$ because $(R \cup \Delta)^* = R^*$ for all relations R.

Let $e, f \in \mathsf{Exp}$. 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.
- $\mathfrak{F} \models e = f$ when $K \models e = f$ holds in every *finite* KA K.
- $\mathfrak{R} \models e = f$ when $\mathcal{P}(S \times S) \models e = f$ for all S.

Kleene algebra

Model theory

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

This talk

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)

We found such a proof — with many ideas inspired by Palka.

Roadmap: Given $e, f \in \mathrm{Exp}$ 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

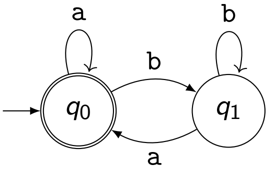
Expressions to automata

Definition

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

- Q is a finite set of *states*; and
- $\rightarrow \subseteq Q \times \Sigma \times Q$ 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 $(q, \mathtt{a}, q') \in \rightarrow$.



Expressions to automata

Definition

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

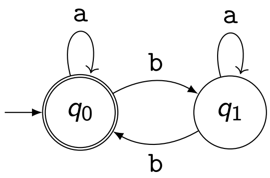
$$\vdash F(q) + \sum_{q \xrightarrow{\mathtt{a}} q'} \mathtt{a} \cdot s(q') \leq s(q) \qquad\qquad F(q) = \begin{cases} 1 & q \in F \\ 0 & q \notin F \end{cases}$$

Example

For the automaton on the right, a solution satisfies

$$\vdash 1 + \mathtt{a} \cdot s(q_0) + \mathtt{b} \cdot s(q_1) \leq s(q_0)$$

$$\vdash 0 + \mathtt{a} \cdot s(q_1) + \mathtt{b} \cdot s(q_0) \leq s(q_1)$$



E.g., $s(q_0) = (\mathtt{a} + \mathtt{b} \cdot \mathtt{a}^* \cdot \mathtt{b})^*$ and $s(q_1) = \mathtt{a}^* \cdot \mathtt{b} \cdot s(q_0)$.

Expressions to automata

Theorem (Kleene 1956; see also Conway 1971)

Every automaton admits a least solution (unique up to equivalence).

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

Lemma (c.f. Kleene 1956; Antimirov 1996; Kozen 2001; Jacobs 2006)

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

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

Automata to monoids

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

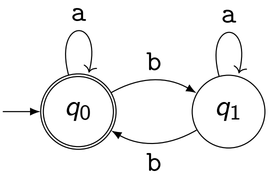
Definition (Transition monoid; McNaughton and Papert 1968)

$(M_A, \circ, \Delta)$ is the monoid where $M_A = \{\xrightarrow{\mathtt{a}_1} \circ \cdots \circ \xrightarrow{\mathtt{a}_n} : \mathtt{a}_1, \ldots, \mathtt{a}_n \in \Sigma\}$.

Example

The transition monoid for the automaton A on the right is carried by $M_A = \{\to_{\mathtt{a}}, \to_{\mathtt{b}}\}$, where

$$\to_{\mathtt{a}} = \{(q_0, q_0), (q_1, q_1)\} \qquad \to_{\mathtt{b}} = \{(q_0, q_1), (q_0, q_1)\}$$



Automata to monoids

Definition (Transition automaton; McNaughton and Papert 1968)

Let $R \in M_A$. 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 (Solving transition automata)

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

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:

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

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 $\hat{L}(e) = \hat{L}(f)$, so $\hat{h}(e) = \hat{h}(f)$. The rest proceeds as before. $\square$

Coq formalization

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

Pomsets

Expressions in *concurrent KA* (CKA) are generated by

$$e, f ::= 0 \mid 1 \mid a \in \Sigma \mid e + f \mid e \cdot f \mid e \parallel f \mid e^* \mid e^\dagger$$

Definition (Bi-KA)

A *bi-KA* is a tuple $(K, +, \cdot, \parallel, {}^*, {}^\dagger, 0, 1)$ where

- $(K, +, \cdot, {}^*)$ and $(K, +, \parallel, {}^\dagger)$ are both KAs, and
- $\parallel$ commutes, i.e., $K \models e \parallel f = f \parallel e$.

A *weak bi-KA* is a bi-KA without the †.

Definition (Concurrent KA)

A *(weak) concurrent KA* is a (weak) bi-KA K satisfying

$$(e \parallel g) \cdot (f \parallel h) \leq (e \cdot f) \parallel (g \cdot h)$$

Pomsets

Example

The *bi-KA of pomset languages* over Σ is $(\mathcal{P}(\mathrm{Pom}(\Sigma)), \cup, \cdot, \parallel, {}^*, {}^\dagger, \emptyset, \{1\})$, where

- $\mathrm{Pom}(\Sigma)$ denotes the set of pomsets over Σ;
- 1 denotes the empty pomset;
- $L \cdot L' = \{U \cdot V : U \in L, V \in L'\}$ and similarly for $\parallel$; and
- $L^* = \{1\} \cup L \cup L \cdot L \cup \cdots$ and $L^\dagger = \{1\} \cup L \cup L \parallel L \cup \cdots$.

Pomsets

Example

The *concurrent KA of pomset ideals* over Σ is $(\mathcal{I}(\Sigma), \cup, \cdot, \|, {}^*, {}^\dagger, \emptyset, \{1\})$, where

- $\mathcal{I}(\Sigma)$ contains the pomset languages downward-closed under $\sqsubseteq$; and
- the operators are as for bi-KA, but followed by downward closure under $\sqsubseteq$.

Pomsets

Theorem (Laurence and Struth 2014)

Let e and f be (weak) concurrent KA expressions.

Now $\mathcal{P}(\mathrm{Pom}(\Sigma)) \models e = f$ if and only if $K \models e = f$ for all (weak) bi-KAs K

Theorem (Laurence and Struth 2017; K., Brunet, Silva, et al. 2018)

Let e and f be weak concurrent KA expressions.

Now $\mathcal{I}(\Sigma) \models e = f$ if and only if $K \models e = f$ for all weak CKAs K

Pomsets

Conjecture

Let e and f be concurrent KA expressions.

Now $\mathcal{I}(\Sigma) \models e = f$ if and only if $K \models e = f$ for all CKAs K

Current techniques do not work!

`<speculation>`

Pomsets

The following roadmap *might* work:

1. Translate CKA expressions to automata

   $\Rightarrow$ Pomset automata (K., Brunet, Luttik, et al. 2019)

   $\Rightarrow$ or HDAs (van Glabbeek 2004; Fahrenberg 2005; Fahrenberg et al. 2022)

2. Translate these automata to *ordered bimonoids* (Bloom and Ésik 1996)

   $\Rightarrow$ see also (Lodaya and Weil 2000; van Heerdt et al. 2021)

3. Translate bimonoids to concurrent KAs.

   $\Rightarrow$ essentially the same recipe?

</speculation>

Further 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*?
- Is there a representation theorem or duality for KA?

`https://kap.pe/slides` `https://kap.pe/papers`

References I

- Antimirov, Valentin M. (1996). "Partial Derivatives of Regular Expressions and Finite Automaton Constructions". In: *Theor. Comput. Sci.* 155.2, pp. 291–319. DOI: 10.1016/0304-3975(95)00182-4.
- Bloom, Stephen L. and Zoltán Ésik (1996). "Free Shuffle Algebras in Language Varieties". In: *Theor. Comput. Sci.* 163.1&2, pp. 55–98. DOI: 10.1016/0304-3975(95)00230-8.
- Conway, John Horton (1971). *Regular Algebra and Finite Machines*. Chapman and Hall, Ltd., London.
- Fahrenberg, Uli (2005). "A Category of Higher-Dimensional Automata". In: *FoSSaCS*, pp. 187–201. DOI: 10.1007/978-3-540-31982-5_12.
- Fahrenberg, Uli et al. (2022). "A Kleene Theorem for Higher-Dimensional Automata". In: *CONCUR*, 29:1–29:18. DOI: 10.4230/LIPIcs.CONCUR.2022.29.

References II

- 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.
- Kappé, Tobias, Paul Brunet, Bas Luttik, et al. (2019). “On series-parallel pomset languages: Rationality, context-freeness and automata”. In: *J. Log. Algebr. Meth. Program.* 103, pp. 130–153. DOI: 10.1016/j.jlamp.2018.12.001.
- Kappé, Tobias, Paul Brunet, Alexandra Silva, et al. (2018). “Concurrent Kleene Algebra: Free Model and Completeness”. In: *ESOP*, pp. 856–882. DOI: 10.1007/978-3-319-89884-1_30.
- 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.

References III

- Kozen, Dexter (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.
- Laurence, Michael R. and Georg Struth (2014). "Completeness Theorems for Bi-Kleene Algebras and Series-Parallel Rational Pomset Languages". In: *RAMiCS*, pp. 65–82. DOI: 10.1007/978-3-319-06251-8_5.
- — (2017). *Completeness Theorems for Pomset Languages and Concurrent Kleene Algebras*. arXiv: 1705.05896.
- Lodaya, Kamal and Pascal Weil (2000). "Series-parallel languages and the bounded-width property". In: *Theor. Comp. Sci.* 237.1, pp. 347–380. DOI: 10.1016/S0304-3975(00)00031-1.
- McNaughton, Robert and Seymour Papert (1968). "The syntactic monoid of a regular event". In: *Algebraic Theory of Machines, Languages, and Semigroups*, pp. 297–312.

References IV

- 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.
- van Glabbeek, Rob J. (2004). "On the Expressiveness of Higher Dimensional Automata: (Extended Abstract)". In: *EXPRESS*, pp. 5–34. DOI: 10.1016/j.entcs.2004.11.026.
- van Heerdt, Gerco et al. (2021). "Learning Pomset Automata". In: *FoSSaCS*, pp. 510–530. DOI: 10.1007/978-3-030-71995-1_26.