Open Universiteit

Institute for Logic, Language and Computation

Completeness and the FMP for KA, revisited

Tobias Kappé

RAMICS, April 4, 2023

Some context

- Program semantics sometimes obey laws of Kleene algebra (KA).

Some context

- Program semantics sometimes obey laws of Kleene algebra (KA).
- What can we (not) prove using these laws?

Some context

- Program semantics sometimes obey laws of Kleene algebra (KA).
- What can we (not) prove using these laws?
- When is something not true by just the laws of KA?

Definition (Kleene algebra)

A *Kleene algebra* is a tuple $(K, +, \cdot, ^*, 0, 1)$ where

- $(K, +, \cdot, 0, 1)$ is an idempotent semiring
- The operator * additionally satisfies

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

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

Kleene algebra

Expressions and equations

Definition

Fix an alphabet Σ. Exp is the set of *regular expressions*, generated by

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

Kleene algebra

Expressions and equations

Definition

Fix an alphabet Σ. 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

$$\hat{h}(0) = 0 \qquad \hat{h}(e + f) = \hat{h}(e) + \hat{h}(f)$$
$$\hat{h}(1) = 1 \qquad \hat{h}(e \cdot f) = \hat{h}(e) \cdot \hat{h}(f)$$
$$\hat{h}(\mathtt{a}) = h(\mathtt{a}) \qquad \hat{h}(e^*) = \hat{h}(e)^*$$

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

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

- $\mathcal{P}(\Sigma^*)$ 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.

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

- $\mathcal{P}(\Sigma^*)$ 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.

Fact: $\mathcal{P}(\Sigma^*) \models e = f$ when e and f denote the same regular language.

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.

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.

Fact: $\mathcal{P}(S \times S) \models (\mathsf{a} + 1)^* = \mathsf{a}^*$ because $(R \cup \Delta)^* = R^*$ for all relations R.

Kleene algebra

Model theory

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

- $\vdash e = f$ when $e = f$ follows from the axioms of KA.

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.

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.

Kleene algebra

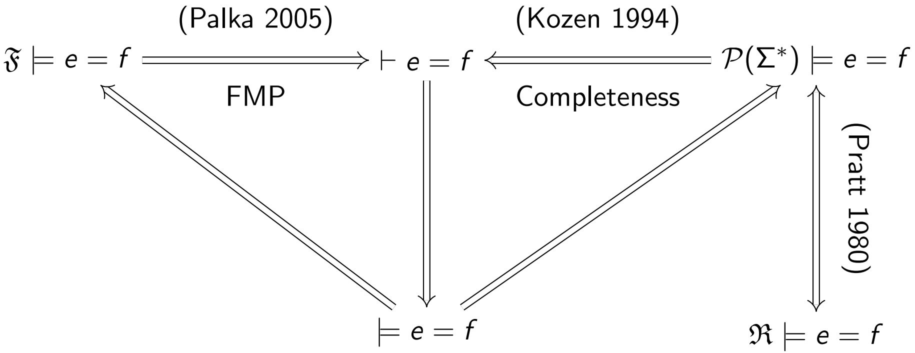
Model theory

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



This paper

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

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

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.

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.

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

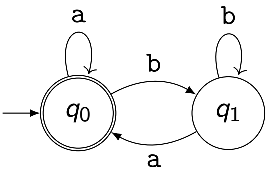
Expressions to automata

Definition

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

Expressions to automata

Definition

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

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

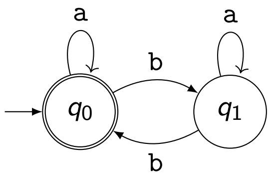
Here, $F(q) = 1$ when $q \in F$ and $F(q) = 0$ otherwise.

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).

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.

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 $A_e = (Q_e, \rightarrow_e, I_e, F_e)$ such that

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

Automata to monoids

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

Definition (Transition monoid; McNaughton and Papert 1968)

$(M_A, \circ, \Delta)$ 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

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

Definition (Transition monoid; McNaughton and Papert 1968)

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

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

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

Automata to monoids

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

Definition (Transition monoid; McNaughton and Papert 1968)

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

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*

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

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 $e, f \in \mathrm{Exp}$, 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 $A_{e,f}$.

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* $\vdash e = f$.

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* $\vdash e = f$.

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 $L(\mathtt{a}) = \{\mathtt{a}\}$.

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.

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 proof assistant.

Coq formalization

- All results formalized in the Coq proof assistant.
- Trusted base:
  - Calculus of Inductive Constructions.
  - Streicher's *axiom K*.
  - Dependent functional extensionality.

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.

Open questions

- Can we apply these ideas to *guarded Kleene algebra with tests*?

Open questions

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

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*?

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?

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.
- 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.

References II

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.

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.