Kleene Algebra — Lecture 1

ESSLLI 2023
Housekeeping

- Best way to reach me is by email.
- Website: https://kap.pe/essl11i.
- 5 lectures, two 40-minute parts, 10-ish minute break.
- Extensive lecture notes, including exercises.
- It is always OK to ask me for clarification.
- It is always OK to discuss the exercises with other people.
These programs are the same... but how do you prove that?
Overview

- We can reason *equationally*, using properties of programs.
- We can reason *operationally*, by comparing abstract machines.
- These are *two sides of the same coin*. 
Equational reasoning

- Speaks to our intuition — you have all done this before.
- Helps to relate *programs* to *specifications*.
- Allows us to prove validity of refactoring operations.
- Solve equations to find program satisfying a specification.
Operational reasoning

- Corresponds much more closely to what computers do.
- Long tradition of powerful automated reasoning.
- Will cover this in more detail from lecture 3 onwards.
Syntax

Primitive actions $\Sigma = \{a, b, c, \ldots\}$.

Compound expressions:

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

Think of $e \in E$ as a pattern of behavior for a program.
Example: Integer Square Root

In a traditional language:

\[
i \leftarrow 0; \\
\text{while } (i + 1)^2 \leq n \text{ do} \\
| \quad i \leftarrow i + 1;
\]

In our language (for now):

\[
\text{init } \cdot (\text{guard } \cdot \text{incr}^*) \cdot \text{validate}
\]
Semantics

Definition (Interpretation)
An *interpretation* is a pair $\langle S, \sigma \rangle$ where $S$ is a set, and $\sigma : \Sigma \rightarrow 2^{S \times S}$.

Definition (Relational semantics)
Let $\sigma$ be an interpretation. $[e]_\sigma$ is a relation on $S$, defined inductively by

$$
\begin{align*}
[0]_\sigma &= \emptyset \\
[1]_\sigma &= \text{id}_S \\
[a]_\sigma &= \sigma(a) \\
[e + f]_\sigma &= [e]_\sigma \cup [f]_\sigma \\
[e \cdot f]_\sigma &= [e]_\sigma \circ [f]_\sigma \\
[e^*]_\sigma &= [e]_\sigma^*
\end{align*}
$$
Example: Integer Square Root

\[
\text{init} \cdot (\text{guard} \cdot \text{incr})^* \cdot \text{validate}
\]

\[
S = \{ f : \{i, n\} \to \mathbb{N} \}
\]

\[
\sigma(\text{init}) = \{ \langle s, s[0/i] \rangle : s \in S \}
\]

\[
\sigma(\text{guard}) = \{ \langle s, s \rangle : (s(i) + 1)^2 \leq s(n) \}
\]

\[
\sigma(\text{incr}) = \{ \langle s, s[s(i) + 1/i] \rangle : s \in S \}
\]

\[
\sigma(\text{validate}) = \{ \langle s, s \rangle : (s(i) + 1)^2 > s(n) \}
\]
Reasoning

Which things are true regardless of $\sigma$?

For instance, $+$ is commutative:

$$[e + f]_\sigma = [e]_\sigma \cup [f]_\sigma = [f]_\sigma \cup [e]_\sigma = [f + e]_\sigma$$

Can you think of any other laws?
Axioms

Definition (Kleene Algebra)
We define $\equiv$ as the smallest congruence on $E$ satisfying the following:

\[
\begin{align*}
e + 0 & \equiv e \\
e + e & \equiv e \\
e + f & \equiv f + e \\
e + (f + g) & \equiv (e + f) + g \\
e \cdot (f \cdot g) & \equiv (e \cdot f) \cdot g \\
e \cdot (f + g) & \equiv e \cdot f + e \cdot g \\
(e + f) \cdot g & \equiv e \cdot g + f \cdot g \\
e \cdot 1 & \equiv e \equiv 1 \cdot e \\
e \cdot 0 & \equiv 0 \equiv 0 \cdot e \\
e + f \cdot g & \leq g \implies f^* \cdot e \leq g \\
e + f \cdot g & \leq f \implies e \cdot g^* \leq f
\end{align*}
\]

Here $e \leq f$ is shorthand for $e + f \leq f$. 
Lemma (Soundness)

If \( e \equiv f \), then \( [e]_\sigma = [f]_\sigma \) for all interpretations \( \sigma \).

Proof sketch.

By induction on the construction of \( \equiv \); for instance, if \( e = g_1 \cdot (g_2 \cdot g_3) \) and \( f = (g_1 \cdot g_2) \cdot g_3 \), then we can derive as follows:

\[
[e]_\sigma = [g_1]_\sigma \circ ([g_2]_\sigma \circ [g_3]_\sigma) = ([g_1]_\sigma \circ [g_2]_\sigma) \circ [g_3]_\sigma = [f]_\sigma \quad \square
\]

Homework exercise: show that if \( [e + f \cdot g]_\sigma \subseteq [g]_\sigma \), then \( [f^* \cdot e]_\sigma \subseteq [g]_\sigma \).
Lemma

If $e \leq f$ and $f \leq e$, then $e \equiv f$.

Proof.

Recall that $e \leq f$ and $f \leq e$ means that $e + f \equiv f$ and $f + e \equiv e$, so

$$e \equiv f + e \equiv e + f \equiv f$$
Reasoning

Lemma
\[ e \cdot (f \cdot e)^* \leq (e \cdot f)^* \cdot e. \]

Proof.
We can first show that
\[ e + ((e \cdot f)^* \cdot e) \cdot (f \cdot e) \leq (e \cdot f)^* \cdot e \]

To see this, we derive that
\[
\begin{align*}
  e + ((e \cdot f)^* \cdot e) \cdot (f \cdot e) &\equiv e + ((e \cdot f)^* \cdot (e \cdot f)) \cdot e \\
  &\equiv (1 + (e \cdot f)^* \cdot (e \cdot f)) \cdot e \\
  &\equiv (e \cdot f)^* \cdot e
\end{align*}
\]
Suppose showing that $e \equiv f$, is not working out.

- Maybe $\llbracket e \rrbracket_\sigma \neq \llbracket f \rrbracket_\sigma$ for a certain (cleverly constructed) $\sigma$.

- Maybe $\llbracket e \rrbracket_\sigma = \llbracket f \rrbracket_\sigma$ for all $\sigma$, but $e \equiv f$ is simply not provable.

How can you tell the difference?

By the end of this course, you will be able to exclude these possibilities.
A language model — motivation

The interpretation $\sigma$ is cumbersome to carry around!

We need a model that is agnostic of the interpretation.

Solution: collect possible sequences of primitive actions.
A language model — ground terms

Definition (Words)
A word over $\Sigma$ is a sequence $a_1 \cdots a_n$ where $a_i \in \Sigma$.
We write $\epsilon$ for the empty word.
When $w, x \in \Sigma^*$, we write $wx$ for the concatenation of $w$ and $x$.

Definition (Languages)
A set of words is called a language. Let $L$ and $K$ be languages.
We write $L \cdot K$ for the language $\{wx : w \in L, x \in K\}$.
We also write $L^*$ for the language $\{w_1 w_2 \cdots w_n : w_i \in L\}$.
Note: this makes $\Sigma^*$ the set of all words.
A language model — definition

Idea: collect all sequences of actions denoted by $e \in \mathbb{E}$ in a language.

Definition (Language model)
We define $[\_ \_]_E : \mathbb{E} \rightarrow 2^{\Sigma^*}$ inductively, as follows:

$$[0]_E = \emptyset$$
$$[1]_E = \{\epsilon\}$$
$$[a]_E = \{a\}$$

$$[e + f]_E = [e]_E \cup [f]_E$$
$$[e \cdot f]_E = [e]_E \cdot [f]_E$$

$$[e^*]_E = [e]_E^*$$
Connecting the models

How do these models interrelate?

**Theorem (Equivalence of models)**

Let \( e, f \in E \). The following are equivalent:

(i) \([e]_E = [f]_E\)

(ii) for all \( \sigma \), \([e]_\sigma = [f]_\sigma\).

**Corollary**

Let \( e, f \in E \). If \( e \equiv f \), then \([e]_E = [f]_E\).
Lemma
Let \( e, f \in \mathbb{E} \). If \( \llbracket e \rrbracket_\mathbb{E} = \llbracket f \rrbracket_\mathbb{E} \), then for all \( \sigma \), we have \( \llbracket e \rrbracket_\sigma = \llbracket f \rrbracket_\sigma \).

Proof sketch.
First, define the action of \( \sigma \) on a language as follows:

\[
\hat{\sigma}(L) = \bigcup_{a_1 \cdots a_n \in L} \sigma(a_1) \circ \cdots \circ \sigma(a_n)
\]

Then, show that if \( g \in \mathbb{E} \), then \( \hat{\sigma}(\llbracket g \rrbracket_\mathbb{E}) = \llbracket g \rrbracket_\sigma \), by induction on \( g \).

Finally, derive \( \llbracket e \rrbracket_\sigma = \hat{\sigma}(\llbracket e \rrbracket_\mathbb{E}) = \hat{\sigma}(\llbracket f \rrbracket_\mathbb{E}) = \llbracket f \rrbracket_\sigma \)
Lemma

Let $e, f \in \mathbb{E}$. If $[e]_{\sigma} = [f]_{\sigma}$ for all $\sigma$, then $[e]_{\mathbb{E}} = [f]_{\mathbb{E}}$.

Proof sketch.

Consider the map $\# : 2^{\Sigma^*} \to 2^{\Sigma^* \times \Sigma^*}$, given by

$$\#(L) = \{ \langle w, wx \rangle : w \in \Sigma^*, x \in L \}$$

One can show that $\#$ is injective. So, it suffices to show that $\#([e]_{\mathbb{E}}) = \#([f]_{\mathbb{E}})$.

Let’s choose $S = 2^{\Sigma^* \times \Sigma^*}$, and set $\sigma(a) = \#(\{a\})$.

Now for $g \in \mathbb{E}$, we have $[g]_{\sigma} = \#([g]_{\mathbb{E}})$.

Finally, derive $\#([e]_{\mathbb{E}}) = [e]_{\sigma} = [f]_{\sigma} = \#([f]_{\mathbb{E}})$.  \qed
Looking ahead

Tomorrow: incorporate reasoning about control flow.