Introduction to Kleene Algebra	**Lecture 13**
CS786 Spring 2004	**March 15, 2004**

Models of KAT

In this lecture we show that the equational theories of KAT, KAT^* (the star-continuous Kleene algebras with tests), and relational Kleene algebras with tests coincide. We also introduce a family of language-theoretic models consisting of regular sets of *guarded strings*, which play the same role in KAT that the regular sets play in Kleene algebra. These results are from [2].

The Language of Kleene Algebra with Tests

Let P and B be disjoint finite sets of symbols. Elements of P are called *primitive actions* and elements of B are called *primitive tests*. *Terms* and *Boolean terms* are defined inductively:

- any primitive action p is a term
- any primitive test b is a Boolean term
- 0 and 1 are Boolean terms
- if p and q are terms, then so are $p+q$, pq, and p^* (suitably parenthesized if necessary)
- if b and c are Boolean terms, then so are $b+c$, bc, and $\overline{b}$ (suitably parenthesized if necessary)
- any Boolean term is a term.

The set of all terms over P and B is denoted $\mathsf{RExp}_{\mathsf{P},\mathsf{B}}$. The set of all Boolean terms over B is denoted $\mathsf{RExp}_{\mathsf{B}}$.

An *interpretation* over a Kleene algebra with tests K is any homomorphism (function commuting with the distinguished operations and constants) defined on $\mathsf{RExp}_{\mathsf{P},\mathsf{B}}$ and taking values in K such that the Boolean terms are mapped to elements of the distinguished Boolean subalgebra.

If K is a Kleene algebra with tests and I is an interpretation over K, we write $K, I \vDash \varphi$ if the formula φ holds in K under the interpretation I according to the usual semantics of first-order logic. We write $\mathsf{KAT} \vDash \varphi$ (respectively, $\mathsf{KAT}^* \vDash \varphi$) if the formula φ is a logical consequence of the axioms of KAT (respectively, KAT^*). The only formulas we consider are equations or equational implications (universal Horn formulas).

We write $\mathsf{KAT} \vDash \varphi$ if the formula φ is a logical consequence of KAT, i.e. if φ holds under all interpretations over Kleene algebras with tests. We write $\mathsf{KAT}^* \vDash \varphi$ if φ holds under all interpretations over star-continuous Kleene algebras with tests.

Guarded Strings

Let P and B be disjoint finite sets of symbols. Our language-theoretic model of Kleene algebras with tests is based on the idea of *guarded strings* over P and B. Guarded strings were introduced in [1].

We obtain a guarded string from a string $x \in \mathsf{P}^*$ by inserting *atoms* interstitially among the symbols of x. An *atom* is a Boolean expression representing an atom (minimal nonzero element) of the free Boolean algebra on generators B.

Formally, an *atom* of $\mathsf{B} = \{b_1, \ldots, b_k\}$ is a string of literals $c_1 c_2 \cdots c_k$, where each $c_i \in \{b_i, \bar{b}_i\}$. This assumes an arbitrary but fixed order $b_1 < b_2 < \cdots < b_k$ on B; for technical reasons, we require the literals in an atom to occur in this order. There are exactly 2^k atoms, and they are in one-to-one correspondence with the truth assignmens to B. We denote atoms of B by $\alpha, \beta, \alpha_0, \ldots$. The set of all atoms of B is denoted $\mathsf{Atoms}_\mathsf{B}$. The set $\mathsf{Atoms}_\mathsf{B}$ will turn out to be the multiplicative identity of our language-theoretic model $\mathsf{Reg}_{\mathsf{P},\mathsf{B}}$.

If $b \in \mathsf{B}$ and α is an atom of B, we write $\alpha \leq b$ if b occurs positively in α and $\alpha \leq \bar{b}$ if b occurs negatively in α. This notation is consistent with the natural order in the free Boolean algebra generated by B.

Intuitively, the symbols of P can be thought of as instructions and atoms as conditions that must be satisfied at some point in the computation. If $\alpha \leq c_i$, then α asserts that c_i holds (and $\bar{c}_i$ fails) at that point in the computation.

Definition 13.1 *A* guarded string *over* P *and* B *is any element of* $(\mathsf{Atoms}_\mathsf{B}\mathsf{P})^*\mathsf{Atoms}_\mathsf{B}$*; that is, any string*

$$\alpha_0 p_1 \alpha_1 p_2 \cdots p_n \alpha_n, \quad n \geq 0,$$

where each α_i *is an atom of* B *and each* $p_i \in \mathsf{P}$*. Note that a guarded string begins and ends with an atom. If* x *is the guarded string above, we define* $\mathsf{first}\, x \stackrel{\text{def}}{=} \alpha_0$ *and* $\mathsf{last}\, x \stackrel{\text{def}}{=} \alpha_n$*. In the case* $n = 0$*,* x *is just a single atom, and* $\mathsf{first}\, x = \mathsf{last}\, x$*.*

The set of all guarded strings over P *and* B *is denoted* $\mathrm{GS}_{\mathsf{P},\mathsf{B}}$*, or just* GS *when* P *and* B *are understood.*

Let $\overline{\mathsf{B}} = \{\bar{b} \mid b \in \mathsf{B}\}$. We denote strings in $(\mathsf{P} \cup \mathsf{B} \cup \overline{\mathsf{B}})^*$, including guarded strings, by the letters $x, y, z, x_1, \ldots$.

The analog of concatenation for guarded strings is *fusion product.*

Definition 13.2 *The* fusion product *operation* $\cdot$ *is a* partial *binary operation on* GS *defined as follows. If* $\mathsf{last}\,x = \mathsf{first}\,y$*, then the fusion product* xy *exists and is equal to the string obtained by concatenating* x *and* y*, but writing the common atom* $\mathsf{last}\,x = \mathsf{first}\,y$ *only once between them.*

For example, if $\mathsf{B} = \{b, c\}$ *and* $\mathsf{P} = \{p, q\}$*, then*

$$bcp\overline{b}c \cdot \overline{b}cq\overline{b}\overline{c} \quad = \quad bcp\overline{b}cq\overline{b}\overline{c}.$$

If $\mathsf{last}x \neq \mathsf{first}y$*, then the fusion product* xy *is undefined. We usually omit the* $\cdot$ *in expressions. If* $A, B \subseteq$ GS*, define*

$$AB \quad \stackrel{\mathrm{def}}{=} \quad \{xy \mid x \in A,\ y \in B,\ xy \ \textit{exists}\}.$$

Thus AB *consists of all existing fusion products of guarded strings in* A *with guarded strings in* B*. For example, if* $\mathsf{B} = \{b, c\}$*,* $\mathsf{P} = \{p, q\}$*, and*

$$\begin{aligned} A &= \{bcp\overline{b}\overline{c}, \overline{b}\overline{c}, bcqb\overline{c}\} \\ B &= \{\overline{b}\overline{c}pbc, \overline{b}\overline{c}, b\overline{c}qbc\}, \end{aligned}$$

then

$$AB \quad = \quad \{bcp\overline{b}\overline{c}pbc, bcp\overline{b}\overline{c}, \overline{b}\overline{c}pbc, \overline{b}\overline{c}, bcqb\overline{c}qbc\}.$$

Whereas the operation $\cdot$ is partial when applied to guarded strings, it is total when applied to *sets* of guarded strings. Note that if there are no existing fusion products of strings from A and B, then $AB = \varnothing$. It is not difficult to show that $\cdot$ is associative, that it distributes over union, and that it has two-sided identity $\mathsf{Atoms}_\mathsf{B}$.

We now define a language-theoretic model $\mathsf{Reg}_{\mathsf{P},\mathsf{B}}$ based on guarded strings. The elements of $\mathsf{Reg}_{\mathsf{P},\mathsf{B}}$ will be the regular sets of guarded strings over P and B (although we have not yet defined *regular* in this context). We will also give a standard interpretation of terms in $\mathsf{RExp}_{\mathsf{P},\mathsf{B}}$ over $\mathsf{Reg}_{\mathsf{P},\mathsf{B}}$ analogous to the standard interpretation of regular expressions as regular sets.

For $A \subseteq$ GS, define inductively

$$A^0 \ \stackrel{\mathrm{def}}{=} \ \mathsf{Atoms}_\mathsf{B} \qquad\qquad A^{n+1} \ \stackrel{\mathrm{def}}{=} \ A \cdot A^n.$$

The asterate operation for sets of guarded strings is defined by

$$A^* \ \stackrel{\mathrm{def}}{=} \ \bigcup_{n \geq 0} A^n.$$

Let $^-$ denote set complementation in $\mathsf{Atoms_B}$. That is, if $A \subseteq \mathsf{Atoms_B}$, then $\overline{A} = \mathsf{Atoms_B} - A$. Consider the structure

$$(2^{\mathrm{GS}},\ 2^{\mathsf{Atoms_B}},\ \cup,\ \cdot,\ ^*,\ ^-,\ \varnothing,\ \mathsf{Atoms_B}),$$

which we denote briefly by 2^{GS}. It is quite straightforward to verify that this is a star-continuous Kleene algebra with tests; that is, it is a model of KAT^*. The Boolean algebra axioms hold for $2^{\mathsf{Atoms_B}}$ because it is a set-theoretic Boolean algebra.

The star-continuity condition follows immediately from the definition of * and the distributivity of fusion product over infinite union. Since

$$B^* \;=\; \bigcup_{n\geq 0} B^n,$$

we have that

$$AB^*C \;=\; A\cdot(\bigcup_{n\geq 0} B^n)\cdot C = \bigcup_{n\geq 0} AB^nC.$$

Both of these expressions denote the set

$$\{xyz \mid x\in A,\ z\in C,\ \exists n\ y\in B^n\}.$$

For $p\in\mathsf{P}$ and $b\in\mathsf{B}$, define

$$G(p) \;\stackrel{\mathrm{def}}{=}\; \{\alpha p\beta \mid \alpha,\beta\in\mathsf{Atoms_B}\} \tag{13.1}$$

$$G(b) \;\stackrel{\mathrm{def}}{=}\; \{\alpha\in\mathsf{Atoms_B} \mid \alpha\leq b\}. \tag{13.2}$$

The structure $\mathsf{Reg_{P,B}}$ is defined to be the subalgebra of 2^{GS} generated by the elements $G(p)$ for $p\in\mathsf{P}$ and $G(b)$ for $b\in\mathsf{B}$. Elements of $\mathsf{Reg_{P,B}}$ are called *regular sets*.

Standard Interpretation

The map G defined on primitive actions and primitive tests in (13.1) and (13.2) extends uniquely by induction to a homomorphism $G : \mathsf{RExp_{P,B}} \to \mathsf{Reg_{P,B}}$:

$$G(p+q) \stackrel{\mathrm{def}}{=} G(p)\cup G(q) \qquad G(pq) \stackrel{\mathrm{def}}{=} G(p)\cdot G(q)$$

$$G(1) \stackrel{\mathrm{def}}{=} \mathsf{Atoms_B} \qquad G(\bar b) \stackrel{\mathrm{def}}{=} \mathsf{Atoms_B} - G(b)$$

$$G(0) \stackrel{\mathrm{def}}{=} \varnothing \qquad G(p^*) \stackrel{\mathrm{def}}{=} G(p)^*.$$

The map G is called the *standard interpretation* over $\mathsf{Reg_{P,B}}$.

Relational Models

Relational Kleene algebras with tests are interesting because they closely model our intuition about programs. In a relational model, the elements of K are binary relations and $\cdot$ is interpreted as relational composition. Elements of the Boolean subalgebra are subsets of the identity relation.

Formally, a *relational Kleene algebra with tests* on a set X is any structure

$$(K,\ B,\ \cup,\ \circ,\ ^*,\ ^-,\ \varnothing,\ \iota)$$

such that

$$(K,\ \cup,\ \circ,\ ^*,\ \varnothing,\ \iota)$$

is a relational Kleene algebra, i.e. K is a family of binary relations on X, $\circ$ is ordinary relational composition, * is reflexive transitive closure, and ι is the identity relation on X; and

$$(B,\ \cup,\ \circ,\ ^-,\ \varnothing,\ \iota)$$

is a Boolean algebra of subsets of ι (not necessarily the whole powerset).

All relational Kleene algebras with tests are star-continuous. We write $\mathsf{REL} \models \varphi$ if the formula φ holds in all relational Kleene algebras in the usual sense of first-order logic.

Completeness of KAT^* over $\mathsf{Reg}_{\mathsf{P},\mathsf{B}}$

Now we show that an equation $p = q$ is a theorem of star-continuous Kleene algebra with tests iff it holds under the standard interpretation G over $\mathsf{Reg}_{\mathsf{P},\mathsf{B}}$, where P and B contain all primitive action and test symbols, respectively, appearing in p and q. Thus $\mathsf{Reg}_{\mathsf{P},\mathsf{B}}$ is the free Kleene algebra with tests on generators P and B. In the next lecture, we will strengthen these results by removing the assumption of star-continuity.

Theorem 13.3 *Let* $p, q \in \mathsf{RExp}_{\mathsf{P},\mathsf{B}}$. *Then*

$$\mathsf{KAT}^* \models p = q \quad \Leftrightarrow \quad G(p) = G(q).$$

Equivalently, $\mathsf{Reg}_{\mathsf{P},\mathsf{B}}$ *is the free star-continuous Kleene algebra with tests on generators* P *and* B.

The forward implication is easy, since $\mathsf{Reg}_{\mathsf{P},\mathsf{B}}$ is a star-continuous Kleene algebra. The converse is a consequence of the following lemma.

Lemma 13.4 *For any star-continuous Kleene algebra with tests K, interpretation $I : \mathsf{RExp}_{\mathsf{P},\mathsf{B}} \to K$, and $p, q, r \in \mathsf{RExp}_{\mathsf{P},\mathsf{B}}$,*

$$I(pqr) \;=\; \sup_{x \in G(q)} I(pxr)$$

where the supremum is with respect to the natural order in K. In particular,

$$I(q) \;=\; \sup_{x \in G(q)} I(x).$$

This result is analogous to the same result for Kleene algebras proved in Lecture **??** and the proof is similar. Note that the star-continuity axiom is a special case.

We are most interested in the second statement, but there is a slight subtlety that requires the stronger first statement as the induction hypothesis. In addition to the existence of the supremum, the more general statement provides a kind of infinite distributivity law over existing suprema. The need for this arises mainly in the induction case for $\cdot$.

Proof of Lemma 13.4. We proceed by induction on the structure of q. The basis consists of cases for primitive tests, primitive actions, 0 and 1. We argue the case for primitive actions and primitive tests explicitly.

For a primitive action $q \in \mathsf{P}$, recall that

$$G(q) \;=\; \{\alpha q \beta \mid \alpha, \beta \in \mathsf{Atoms}_\mathsf{B}\}.$$

Then

$$\begin{aligned} I(pqr) &= I(p)I(1)I(q)I(1)I(r) \\ &= \sup\{I(p)I(\alpha)I(q)I(\beta)I(r) \mid \alpha, \beta \in \mathsf{Atoms}_\mathsf{B}\} \\ &= \sup\{I(p\alpha q\beta r) \mid \alpha, \beta \in \mathsf{Atoms}_\mathsf{B}\} \\ &= \sup\{I(pxr) \mid x \in G(q)\}. \end{aligned}$$

Finite distributivity was used in the second step.

For a primitive test $b \in B$, recall that

$$G(b) \;=\; \{\alpha \mid \alpha \le b\}.$$

Then

$$\begin{aligned} I(pbr) &= I(p)I(b)I(r) \\ &= \sup\{I(p)I(\alpha)I(r) \mid \alpha \le b\} \\ &= \sup\{I(p\alpha r) \mid \alpha \le b\} \\ &= \sup\{I(pxr) \mid x \in G(b)\}. \end{aligned}$$

Again, finite distributivity was used in the second step.

The induction step consists of cases for $+$, $\cdot$, *, and $^-$. The cases other than $\cdot$ and $^-$ are the same as in the proof of Theorem **??** of Lecture **??**.

For the case $\cdot$, recall that

$$G(qq') = G(q) \cdot G(q') = \{y\alpha z \mid y\alpha \in G(q),\ \alpha z \in G(q')\}.$$

Applying the induction hypothesis twice,

$$\begin{aligned} I(pqq'r) &= \sup\{I(pqvr) \mid v \in G(q')\} \\ &= \sup\{\sup\{I(puvr) \mid u \in G(q)\} \mid v \in G(q')\} \\ &= \sup\{I(puvr) \mid u \in G(q),\ v \in G(q')\}. \end{aligned}$$

The last step follows from a purely lattice-theoretic argument: if all the suprema in question on the left hand side exist, then the supremum on the right hand side exists and the two sides are equal.

Now

$$\begin{aligned} &\sup\{I(puvr) \mid u \in G(q),\ v \in G(q')\} \\ &\quad = \sup\{I(py\alpha\beta zr) \mid y\alpha \in G(q),\ \beta z \in G(q')\} \\ &\quad = \sup\{I(py\alpha\alpha zr) \mid y\alpha \in G(q),\ \alpha z \in G(q')\} \qquad (13.3) \\ &\quad = \sup\{I(py\alpha zr) \mid y\alpha \in G(q),\ \alpha z \in G(q')\} \\ &\quad = \sup\{I(pxr) \mid x \in G(qq')\}. \end{aligned}$$

The justification for step (13.3) is that if $\alpha \neq \beta$, then the product in K is 0 and does not contribute to the supremum.

For the case $^-$, recall that

$$G(\bar{b}) = \mathsf{Atoms_B} - G(b) = \{\alpha \mid \alpha \not\leq b\} = \{\alpha \mid \alpha \leq \bar{b}\}.$$

Then

$$I(p\bar{b}r) = \sup\{I(p\alpha r) \mid \alpha \leq \bar{b}\} = \sup\{I(p\alpha r) \mid \alpha \in G(\bar{b})\}.$$

□

Proof of Theorem 13.3. If $\mathsf{KAT}^* \models p = q$ then $G(p) = G(q)$, since $\mathsf{Reg_{P,B}}$ is a star-continuous Kleene algebra with tests. Conversely, if $G(p) = G(q)$, then by Lemma 13.4, for any star-continuous Kleene algebra with tests K and any interpretation I over K, $I(p) = I(q)$. Therefore $\mathsf{KAT}^* \models p = q$. □

Completeness over Relational Models

Finally we show completeness of KAT^* over relational interpretations. It will suffice to construct a relational model isomorphic to $\mathsf{Reg}_{\mathsf{P},\mathsf{B}}$. This construction is similar to a construction we have seen before for Kleene algebra in Lecture **??** for regular sets.

For A any set of guarded strings, define

$$h(A) \stackrel{\text{def}}{=} \{(x, xy) \mid x \in \text{GS},\ y \in A\}.$$

Lemma 13.5 *The language-theoretic model* 2^{GS} *and its submodel* $\mathsf{Reg}_{\mathsf{P},\mathsf{B}}$ *are isomorphic to relational models.*

Proof. We show that the function $h : 2^{\text{GS}} \to 2^{\text{GS}\times\text{GS}}$ defined above embeds 2^{GS} isomorphically onto a subalgebra of the Kleene algebra of all binary relations on GS.

It is straightforward to verify that h is a homomorphism:

$$\begin{aligned}
h(A \cup B) &= h(A) \cup h(B) \\
h(AB) &= \{(z, zr) \mid z \in \text{GS},\ r \in AB\} \\
&= \{(z, zpq) \mid z \in \text{GS},\ p \in A,\ q \in B\} \\
&= \{(z, zp) \mid z \in \text{GS},\ p \in A\} \\
&\qquad \circ \{(zp, zpq) \mid z \in \text{GS},\ p \in A,\ q \in B\} \\
&= \{(z, zp) \mid z \in \text{GS},\ p \in A\} \circ \{(y, yq) \mid y \in \text{GS},\ q \in B\} \\
&= h(A) \circ h(B). \\
h(A^*) &= h(\bigcup_{n \geq 0} A^n) \\
&= \bigcup_{n \geq 0} h(A)^n \\
&= h(A)^* \\
h(\mathsf{Atoms}_\mathsf{B}) &= \{(x, x\alpha) \mid x \in \text{GS}, \alpha \in \mathsf{Atoms}_\mathsf{B}\} \\
&= \{(x, x) \mid x \in \text{GS}\} \\
&= \iota \\
h(0) &= \varnothing \\
h(\overline{B}) &= h(\{\alpha \mid \alpha \notin B\}) \\
&= \{(x, x\alpha) \mid \alpha \notin B\} \\
&= \{(y\alpha, y\alpha) \mid \alpha \notin B\} \\
&= \iota - \{(y\alpha, y\alpha) \mid \alpha \in B\} \\
&= \iota - h(B).
\end{aligned}$$

The function h is injective, since A can be uniquely recovered from $h(A)$:

$$A \quad = \quad \{y \mid \exists \alpha \ (\alpha, y) \in h(A)\}.$$

The submodel $\mathsf{Reg}_{\mathsf{P,B}}$ is perforce isomorphic to a relational model on GS, namely the image of $\mathsf{Reg}_{\mathsf{P,B}}$ under h. □

Combining Theorem 13.3, Lemma 13.5, and the fact that all relational models are star-continuous Kleene algebras with tests, we have

Theorem 13.6 *Let* REL *denote the class of all relational Kleene algebras with tests. Let* $p, q \in \mathsf{RExp}_{\mathsf{P,B}}$. *The following are equivalent:*

(i) $\mathsf{KAT}^* \vDash p = q$

(ii) $G(p) = G(q)$

(iii) $\mathsf{REL} \vDash p = q$.

In the next lecture we will remove the assumption of star-continuity and show that the statement $\mathsf{KAT} \vDash p = q$ can be added to this list. Thus KAT is complete for the equational theory of relational models and $\mathsf{Reg}_{\mathsf{P,B}}$ forms the free KAT on generators P and B. This result is analogous to the completeness result of Lecture **??**, which states that the regular sets over a finite alphabet P form the free Kleene algebra on generators P.

References

[1] Donald M. Kaplan. Regular expressions and the equivalence of programs. *J. Comput. Syst. Sci.*, 3:361–386, 1969.

[2] Dexter Kozen and Frederick Smith. Kleene algebra with tests: Completeness and decidability. In D. van Dalen and M. Bezem, editors, *Proc. 10th Int. Workshop Computer Science Logic (CSL'96)*, volume 1258 of *Lecture Notes in Computer Science*, pages 244–259, Utrecht, The Netherlands, September 1996. Springer-Verlag.