COMPSCI 250: Introduction to Computation

Lecture #29: Proving Regular Language Identities

David Mix Barrington and Mordecai Golin

17 April 2024

Regular Language Identities

- Regular Language Identities
- The Semiring Axioms Again
- Identities Involving Union and Concatenation
- Proving the Distributive Law
- The Inductive Definition of Kleene Star
- Identities Involving Kleene Star
- $(ST)^*$, S^*T^*, and $(S + T)^*$

Languages From Number Theory

- We can easily make a regular expression for the set of even-length strings of a's, "$(aa)^*$", or the odd-length strings of a's, "$(aa)^*a$", or the set of strings of a's whose length is congruent to 3 modulo 7, "$a^3(a^7)^*$", or the set of strings whose length is congruent to 1, 2, or 5 modulo 6, "$(a + a^2 + a^5)(a^6)^*$".

- What about the set of strings over {a,b} that have an even number of a's? A good first guess is that such a string is a concatenation of zero or more strings, each of which has exactly two a's. This would be the language $(b^*ab^*ab^*)^*$.

Languages From Number Theory

- But this isn't exactly right, because "bb", for example, has 0 a's and 0 is even. A correct expression for this language is $(b + ab^*a)^*$ -- we can divide any such string into pieces which either have exactly two a's (with some number of b's between) or are just b's themselves.
- It's harder to get the strings with a number of a's congruent to 3 mod 7, or the strings with an even number of a's *and* an even number of b's, but both are possible.

Regular Expression Identities

- In this lecture and the next we'll use our new formal definition of the regular languages to prove things about them.
- In particular, in this lecture we'll prove a number of **regular language identities**, which are statements about languages where the types of the free variables are "regular expression" and which are true for all possible values of those free variables.

Regular Expression Identities

- For example, if we view the union operator + as "addition" and the concatenation operator · as "multiplication", then the rule $S(T + U) = ST + SU$ is a statement about languages and (as we'll prove) is a regular language identity. In fact it's a language identity as regularity doesn't matter.
- We can use the inductive definition of regular expressions to prove statements about the whole family of them -- this will be the subject of the next lecture.

The Semiring Axioms Again

- The set of natural numbers, with the ordinary operations $+$ and $\times$, forms an algebraic structure called a **semiring**.

- Earlier we proved the semiring axioms for the naturals from the Peano axioms and our inductive definitions of $+$ and $\times$.

- It turns out that the languages form a semiring under union and concatenation, and the regular languages are a **subsemiring** because they are **closed** under $+$ and $\cdot$. That is, if R and S are regular, so are $R + S$ and $R \cdot S$.

The Semiring Axioms Again

- Both operations of a semiring must be associative and each must have an identity. For languages, $\emptyset$ is the identity for union and $\{\lambda\} = \emptyset^*$ is the identity for concatenation, as $\emptyset + R = R + \emptyset = R$ and $R\emptyset^* = \emptyset^*R = R$.
  We also need the distributive law which we'll prove soon.

- Note that $+$ is commutative but $\cdot$ is not as in general XY and YX are different languages. There are other identities like $X + X = X$ (addition is *idempotent*) that are not true for the natural numbers.

Clicker Question #1

- Consider the rule "$(X + Y)^2 = X^2 + Y^2$", where squaring denotes multiplying an element by itself in the semiring S. Which of these statements is *true*?
- (a) The rule is never true for any semiring.
- (b) The rule is always true if $XY+YX = 0$ (unless $X = Y$), that is, if multiplication is **anticommutative**).
- (c) The rule is true when $X \neq Y$ if $XY+YX = 0$.
- (d) The rule is always true if the multiplication operation is commutative.

Not the Answer

Clicker Answer #1

- Consider the rule "$(X + Y)^2 = X^2 + Y^2$", where squaring denotes multiplying an element by itself in the semiring S. Which of these statements is *true*?

works for {0, 1}

- (a) The rule is never true for any semiring.
- (b) The rule is always true if $XY+YX = 0$ (unless $X = Y$), that is, if multiplication is **anticommutative**).

If $X=Y$, RHS $= X^2+X^2+X^2+X^2$ which might or might not equal X^2+X^2

- *(c) The rule is true when $X \neq Y$ if $XY+YX = 0$.*

$(X+Y)(X+Y) = XX+XY+YX+YY = XX + YY$

- (d) The rule is always true if the multiplication operation is commutative.

could fail easily

(b) versus (c)?

- Consider the rule "$(X + Y)^2 = X^2 + Y^2$", where squaring denotes multiplying an element by itself in the semiring S. Which of these statements is *true*?

The statements (b) and (c) look the same, but (b) says that the $(X+Y)^2=X^2+Y^2$ rule *also* works for X=Y, which might not be true.

- (b) The rule is always true if $XY+YX = 0$ (unless $X = Y$), that is, if multiplication is **anticommutative**).

If X=Y, RHS $= X^2+X^2+X^2+X^2$ which might or might not equal X^2+X^2

- *(c) The rule is true when $X \neq Y$ if $XY+YX = 0$.*

$(X+Y)(X+Y) = XX+XY+YX+YY = XX + YY$

Union and Concatenation

- We've already proved everything we need to know about identities that just use + for languages, since they are **set identities** for the union operator.
- We know that:
  $S + T = T + S$
  $S + (T + U) = (S + T) + U$
  $S + \emptyset = \emptyset + S = S,$
  $S + S = S$
  $S + \Sigma^* = \Sigma^*.$

Union and Concatenation

- We looked at concatenation of languages back in Chapter 2 of the textbook.
- Statements like $S(TU) = (ST)U$, $S\emptyset = \emptyset S = \emptyset$, and $S\emptyset^* = \emptyset^* S = S$ may be proved by the equational sequence method.
- To prove "$X = Y$", for example, we let w be an arbitrary string and prove $w \in X \leftrightarrow w \in Y$.

Union and Concatenation

- For example, $w \in (ST)U \leftrightarrow$ $\exists u{:}\exists z{:}(w = uz) \wedge (u \in ST) \wedge (z \in U) \leftrightarrow$ $\exists x{:}\exists y{:}\exists z{:}(w = xyz) \wedge (x \in S) \wedge (y \in T) \wedge (z \in U)$ $\leftrightarrow \exists x{:}\exists v{:}(w = xv) \wedge (x \in S) \wedge (v \in TU) \leftrightarrow$ $w \in S(TU)$.
- At each stage we use the definition of concatenation of languages or the associativity of concatenation of strings, "$x(yz) = (xy)z$", which we've already proved.

Proving the Distributive Law

- The equational sequence method also works to prove S(T + U) = ST + SU, using our definitions and some logical rules.

$$w \in S(T + U) \leftrightarrow$$

$$\exists u{:}\exists v{:}(w = uv) \wedge u \in S \wedge v \in (T + U) \leftrightarrow$$

$$\exists u{:}\exists v{:}\ w = uv \wedge u \in S \wedge (v \in T \vee v \in U) \leftrightarrow$$

$$\exists u{:}\exists v{:}\ w = uv \wedge [(u \in S \wedge v \in T) \vee (u \in S \wedge v \in U)] \leftrightarrow$$

$$(\exists u{:}\exists v{:}w = uv \wedge u \in S \wedge v \in T) \vee (\exists u{:}\exists v{:}w = uv \wedge u \in S \wedge v \in U)$$

$$\leftrightarrow$$

$$w \in ST \vee w \in SU \leftrightarrow$$

$$w \in ST + SU$$

The Inductive Definition of Star

- To prove identities about the Kleene star operation, we use its inductive definition.
- If A is any language, we define A^* by three rules:
- (1) $\lambda \in A^*$,
- (2) if $u \in A^*$ and $v \in A$, then $uv \in A^*$, and
- (3) a string is only in A^* if it can be proved to be so by rules (1) and (2).

The Inductive Definition of Star

- The definition we gave earlier, "$w \in A^*$ if and only if w is the concatenation of zero or more strings, each of which is in A" is equivalent.
- By induction on naturals n, we can prove that any concatenation of n strings from A is in A^* according to the second definition.
- And we can prove by induction on all strings w in A^* (according to the second definition) that there exists an n such that w is the concatenation of n strings from A.

Clicker Question #2

- Let $\Sigma = \{a, b\}$.
  Let $P(w)$, for $w \in \Sigma^*$, be "w does not end in aa or bb". Let X denote the language $(ab + ba)^*$.
  In proving "$\forall w: (w \in X) \rightarrow P(w)$", what's the base case of the induction?
- (a) $P(0)$
- (b) $P(\lambda)$
- (c) $P(ab) \wedge P(ba)$
- (d) $\forall v: P(v) \rightarrow (P(vab) \wedge P(vba))$

Not the Answer

Clicker Answer #2

- Let $\Sigma = \{a, b\}$.
  Let P(w), for $w \in \Sigma^*$, be "w does not end in aa or bb".
  Let X denote the language $(ab + ba)^*$.
  In proving "$\forall w: (w \in X) \to P(w)$", what's the base case of the induction?
- (a) P(0) (wrong type)
- *(b) P(λ)*
- (c) $P(ab) \wedge P(ba)$ (misses case of P(λ))
- (d) $\forall v: P(v) \to (P(vab) \wedge P(vba))$ (inductive step)

Structural Induction

- This is an example of a general phenomenon -- any of our **structural inductions** on the definition of a class could be rephrased as inductions on the naturals.
- Rather than proving $P(w)$ for all strings w, for example, we could let $Q(n)$ mean "$P(w)$ for all w of length n" and then prove $Q(n)$ for all naturals n. The proof of $Q(n) \rightarrow Q(n+1)$ would essentially be the same as the proof of $P(w) \rightarrow P(wa)$.

Identities for Kleene Star

- The statement "($u \in A^* \wedge v \in A^*$) $\rightarrow uv \in A^*$", or "A^* is closed under concatenation", is *not* part of the definition of Kleene star.
- It looks very much like our rule (2) which says "($u \in A^* \wedge v \in A$) $\rightarrow uv \in A^*$", but it requires a proof.
- Let's prove this closure rule by induction on all strings v in A^*.

A^* Closed Under Concatenation

- Our statement $P(v)$ is "$u \in A^* \rightarrow uv \in A^*$", where we have let u be arbitrary.
- The base case is $v = \lambda$, and it is clear that if $u \in A^*$ and $v = \lambda$, then $uv \in A^*$ since $uv = u$.
- For the induction, assume that $v = wx$, that $w \in A^*$, that $x \in A$, and that we already know $P(w)$, which says that $u \in A^* \rightarrow uw \in A^*$.

A^* Closed Under Concatenation

- Now to prove $P(v)$, we assume $u \in A^*$, derive $uw \in A^*$ from the IH, and derive that $uv = uwx$ is in A^*.
- This follows from rule (2), because $uw \in A^*$ and $x \in A$.
- This should remind you of the proof that the path relation on graphs is transitive, using the inductive definition of paths.

$(ST)^*$, S^*T^*, and $(S + T)^*$

- It is generally much easier to prove subset relationships than set equalities from the Kleene star definition.
- Equality identities with the Kleene star, like $(S^*)^* = S^*$ are most easily proved by showing both directions, here $(S^*)^* \subseteq S^*$ and $S^* \subseteq (S^*)^*$.
- These in turn follow from the identities $T \subseteq T^*$ and $(S \subseteq T) \rightarrow (S^* \subseteq T^*)$. The second of these follows from $(S \subseteq T^*) \rightarrow (S^* \subseteq T^*)$.

$(ST)^*$, S^*T^*, and $(S + T)^*$

- How shall we prove that $S \subseteq T^* \rightarrow S^* \subseteq T^*$?
- We'll assume $S \subseteq T^*$, let $P(w)$ be "$w \in T^*$", and prove $P(w)$ for all w in S^*.
- For the base case, $w = \lambda$ and we know $\lambda \in T^*$.
- For the induction, assume $w = xy$ with $P(x)$ true and $y \in S$. So $x \in T^*$ by the IH, $y \in T^*$ because $S \subseteq T^*$, and then $w = xy$ is in T^* by the closure of T^* under concatenation.

$(ST)^*$, S^*T^*, and $(S + T)^*$

- We have seen that parentheses matter, so that $(ST)^*$ and S^*T^* are two different languages for most choices of S and T.
- (We saw that $(ab)^* \neq a^*b^*$, for example.)
- But we can prove that both $(ST)^*$ and S^*T^* are contained in $(S + T)^*$, using the identities above.

Clicker Question #3

- Let S and T be any regular expressions. Which of these statements *must be* true?
- (a) $(S^*T + TS^*)^* = (S + T)^*$
- (b) $((S + T^*)(T + S^*))^* = (S + T)^*$
- (c) $(ST^*)^* = (S + T)^*$
- (d) $(ST+TS)^* = (S + T)^*$

Not the Answer

Clicker Answer #3

- Let S and T be any regular expressions. Which of these statements *must be* true?
- (a) $(S^*T + TS^*)^* = (S + T)^*$ (LHS misses S)
- *(b)* $((S + T^*)(T + S^*))^* = (S + T)^*$ (has S, T)
- (c) $(ST^*)^* = (S + T)^*$ (LHS misses T)
- (d) $(ST+TS)^* = (S + T)^*$ (LHS misses S, T)

Why is (b) True?

- (b) $((S + T^*)(T + S^*))^* = (S + T)^*$
- The RHS is the set of all strings of S's and T's.
- We need to show that the expression inside the last star of the LHS contains both S and T.
- But we can make S from SS^* and we can make T from T^*T, and each of these are both part of the options for $((S + T^*)(T + S^*))^*$.