Lecture 9 The Myhill-Nerode Theorem

Thursday, February 4, 2021 10:58 AM

Σ : alphabet finite set of symbols or letters

Σ^* : collection of all strings

an operation <u>concatenation</u>

$$\underbrace{x}_{\in\Sigma^*} \cdot \underbrace{y}_{\in\Sigma^*} = \underbrace{xy}_{\in\Sigma^*}$$

a <u>binary</u> operation

a unit ε $\quad x \cdot \varepsilon = \varepsilon \cdot x = x$

$x \cdot (y \cdot z) = (x \cdot y) \cdot z$ ASSOCIATIVITY

A <u>monoid</u>

Another example S : finite set

$S \to S$ all functions from S to itself

This is a monoid

operation is function composition

Equivalence relations on a set define a <u>partition</u> of the set into equivalence classes. The <u>index</u> of an eq. rel is the number of equivalence classes.

Let's consider eq. rel. on Σ^* and see how concatenation interacts with the concept.

<u>Def</u> An equivalence relation R on Σ^* is said to be right invariant if whenever $x R y$ then $\forall z \in \Sigma^*$ we have

$$xz \; R \; yz$$

Suppose we have a DFA

$$M = (S, \Sigma, s_0, \delta, F)$$

$$\delta^* : S \times \Sigma^* \to S$$

$$\delta^*(s, xy) = \delta^*(\delta^*(s, x), y)$$

<u>Def</u> $x R_M y$ if and only if $\delta^*(s_0, x) = \delta^*(s_0, y)$

<u>FACT</u> This is an example of a right invariant relation.

<u>def</u> $L \subseteq \Sigma^*$, L not <u>necessarily</u> regular. Define R_L

$x R_L y$ iff $\forall z \quad xz \in L \Leftrightarrow yz \in L$.

FACT This is also right invariant.

THEOREM (Myhill-Nerode)

The following are equivalent:

1. The language L is accepted by a DFA (i.e. L is regular)
2. L is the union of <u>some</u> of the equivalence classes of <u>some</u> right invariant equivalence relation of finite index
3. The equivalence relation R_L has finite index. Any relation satisfying (2) will <u>refine</u> R_L.

<u>PROOF</u> (1) $\Rightarrow$ (2)

$$M = (S, s_0, \delta, F)$$

We know R_M is right-invariant.

How many equivalence classes does R_M have?

Ans: one for every state

$q \in S \quad S_q := \{x \mid \delta^*(s_0, x) = q\}$

$\therefore R_M$ has finite index.

$$L = \bigcup_{q \in F} S_q$$

(2) $\Rightarrow$ (3) $\quad R_1$ <u>refines</u> R_2 means

if $x R_1 y$ then $x R_2 y$.

Let R be any right-invariant equivalence relation of <u>finite</u> index such that L is the <u>union</u> of some of the equivalence classes of R.

Suppose $x R y$ $\qquad x, y \in \Sigma^*$

Suppose $xz \in L$ then $yz \in L$. Why?

$xz R yz$ since R is right invariant so xz, yz are in one equivalence class of R. That equivalence class is a subset of L so

$$xz \in L \Rightarrow yz \in L$$

we can reverse this easily

$$yz \in L \Rightarrow xz \in L$$

ie $\quad xz \in L \Leftrightarrow yz \in L \quad \forall z \in \Sigma^*$

But this <u>means</u> $x R_L y$

So $\quad x R y \Rightarrow x R_L y$.

(3) $\Rightarrow$ (1) We construct a m/c from R_L

$$M' = (S', s_0', \delta', F')$$

S' : the equivalence classes of R_L

$$s_0' = [\varepsilon]$$

$$\delta'([x], a) = [xa]$$

$$F' = \{[x] \mid x \in L\}$$

EXERCISE <u>for you</u> : Prove that the $L(M') = L$.

END of the PROOF

ISOMORPHISM of MACHINES

$M_1 = (S_1, s_1, \delta_1, F_1) \quad M_2 = (S_2, s_2, \delta_2, F_2)$

We say M_1 and M_2 are <u>isomorphic</u> if there is a function $\varphi : S_1 \to S_2$ such that:

(1) φ is a bijection

(2) $\varphi(\delta_1(s, a)) = \delta_2(\varphi(s), a)$

$$\begin{tikzcd}
s \arrow[r, "a"] \arrow[d] & \delta_1(s,a) \arrow[d] \\
& \varphi(\delta_1(s,a)) \arrow[d, phantom, "\shortparallel"] \\
\varphi(s) \arrow[r, "a"'] & \delta_2(\varphi(s), a)
\end{tikzcd}$$

(3) $s \in F_1$ iff $\varphi(s) \in F_2$.

It is easy to see that if we have such an isomorphism $L(M_1) = L(M_2)$.

<u>PROP</u> The machine constructed in the last part of the MN Thm proof is the <u>unique</u> (up to isomorphism) minimal machine recognizing L.

With NFA's you can have distinct minimal versions. There is an algorithm for finding a minimal NFA but it is very complex & expensive.