Lecture 7

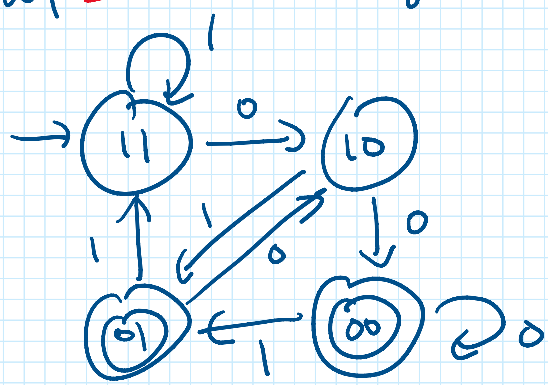
Tuesday, 16 February, 2021 10:33

Today: fooling sets
proving non-regularity

Ex from previous lectures

$L_2 = \{ w \mid 2\text{nd to last symbol in } w \text{ is } 0 \}$

DFA:



4 states turns out to be optimal.
cannot get smaller DFA (why? ~~later~~ today)

state label is last two symbols seen (1 means not 0)

Idea: to prove every DFA for L needs $\geq k$ states
show that there are $\geq k$ different memory configs we need to distinguish. otherwise get "confused" or "fooled"

consider the set $F = \{00, 01, 10, 11\}$

Claim: in any DFA for L_2

$$\delta^*(s, x) \neq \delta^*(s, y) \quad \text{for } x, y \in F \text{ s.t. } x \neq y.$$

Suppose (towards contradiction)

that $\delta^*(s, 00) = \delta^*(s, 01)$

this implies $\delta^*(s, 000) = \delta^*(s, 010)$

this is because

this is because

$$\delta^*(s, 000) = \delta(\delta^*(s,00), 0)$$
$$= \delta(\delta^*(s, 01), 0)$$
$$= \delta^*(s, 010)$$

$000 \in L_2$ $\qquad$ $010 \notin L_2$

$\rightarrow \delta^*(s,000) \in A$ $\qquad$ $\delta^*(s,010) \notin A$.

$\rightarrow \delta^*(s,000) \neq \delta^*(s,010)$

contradiction! $\qquad \delta^*(s,00) \neq \delta^*(s,01)$.

$\delta^*(s, 00\varepsilon) \neq \delta^*(s, 10\varepsilon)$ because $00 \in L$, $10 \notin L$

$\delta^*(s, 00\varepsilon) \neq \delta^*(s, 11\varepsilon)$

For every pair $x, y \in F$ $\quad x \neq y$

$\exists z \quad xz \in L \quad yz \notin L$. (or vice versa)

(e.g. if $x = 00$, $y = 01$, take $z = 0$
If $x = 00$, $y = 10$ take $z = \varepsilon$)

in all cases $\delta^*(s,x) \neq \delta^*(s,y)$

need to remember at least 4 possible states.

this proves no DFA for L_2 w/ < 4 states.

Generalize idea $\rightarrow$ Fooling Sets.

Given language L, a fooling set F for L

is a set of strings s.t.

$\forall\, x, y \in F$ s.t. $x \neq y$.

$\exists\, z \in \Sigma^*$ s.t. either $xz \in L$ & $yz \notin L$

or $xz \notin L$ & $yz \in L$.

true for all DFAs for L.

$$\delta^*(s,x) = \delta^*(s,y) \Rightarrow \delta^*(s,xz) = \delta^*(s,yz).$$

if $xz \in L$ & $yz \notin L$,

$\delta^*(s, xz) \in A$ & $\delta^*(s, yz) \notin A$

$$\Rightarrow \delta^*(s,xz) \neq \delta^*(s,yz).$$

$$\Rightarrow \delta^*(s,x) \neq \delta^*(s,y).$$

Fooling set F for L:

each $x \in F$ has its own $\delta^*(s,x)$

that's different from $\delta^*(s,y)$ for all other $y \in F$.

If I give you fooling set F,

<u>every</u> DFA for L has at least $|F|$ states.

$L_k = \{ w \mid k\text{-th to last symbol in } w \text{ is } 0 \}$

Claim $\{0,1\}^k$ = set of length k binary strings

is a fooling set for L_k

is a fooling set for L_k

if x & y differ in position i from the end

pad out w/ $z = 0^{k-i}$

xz & yz now differ in the k-th to last position

$\rightarrow$ every DFA for L_k requires $\geq |\{0,1\}^k| = 2^k$ states.

$\rightarrow$ $\exists$ NFA w/ $k+1$ states.

Let's take this further.

We said if F is a fooling set for L, every DFA for L requires $\geq |F|$ states.

what if F is infinite?

It's not regular! (need a DFA w/ ∞ states contradicts the $\exists$ in DFA)

To prove a language not regular, show $\exists$ infinite fooling set

every finite subset of this infinite fooling set is still a fooling set

$\emptyset$ is always fooling

infinite fooling set $\Rightarrow$ not regular

finite fooling set $\not\Rightarrow$ regular.

example from before

Claimed: $L = \{ 0^n 1^n \mid n \in \mathbb{N} \}$ is not regular.

intuition: keep track of how many 0's.
distinguish between ∞ many 0^n's.

informs fooling set! Antagonist (Patrick, grader) picks L.

$F = \{ 0^n \mid \underline{n \in \mathbb{N}} \}$. You (student) pick F.

F is infinite.

Let $x, y \in F$ $x \neq y$. $\Rightarrow x = 0^i$ $y = 0^j$ $i \neq j$.

Antagonist picks $x, y \in F$.

need to find z so that $xz \in L$ & $yz \notin L$ (or vice versa)

You pick z.

set $z = 1^i$ $xz = 0^i 1^i \in L$

$yz = 0^j 1^i \notin L$

F is infinite. So L is not regular.

Antagonist is convinced.

another example:

Palindromes $= \{ w \mid w = w^R \}$ $racecar^R = racecar$

intuition: keep track of first half of the input
in order to verify that second half is reverse of the first half.

$F = \{ 0^n 1 \mid \underline{n \in \mathbb{N}} \}$.

F is infinite

let $x, y \in F$ s.t. $x \neq y$. $x = 0^i 1$ & $y = 0^j 1$ where $i \neq j$

set $z = x^R = 10^i$ $xz = xx^R = 0^i 1 1 0^i$

set $z = x^R = 1 0^i$

$xz = xx^R = 0^i 1 1 0^i$

$(xz)^R = (xx^R)^R = xx^R = 0^i 1 1 0^i$

$xz \in$ palindromes

$yz = 0^j 1 1 0^i \notin$ palindromes

but F is <u>infinite</u>. palindromes is not regular.

example: $L_{0=1} = \{ w \mid \#(0,w) = \#(1,w) \}$.

just like in $\{ 0^n 1^n \mid n \in \mathbb{N} \}$

$F = \{ 0^n \mid n \in \mathbb{N} \}$ is a fooling set.

if $x = 0^i$ $y = 0^j$ $z = 1^i$ is still "distinguishing suffix"

since $\#(0, xz) = \#(1, xz) = i$

$\#(0, yz) = j \neq i = \#(1, yz)$.

equivalently: observe that $\{ 0^n 1^n \mid n \in \mathbb{N} \} = L_{0=1} \cap L(0^* 1^*)$

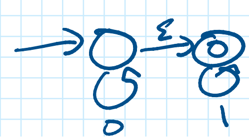
closure property of reg langs if L_1 & L_2 reg

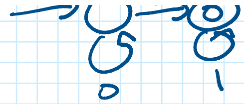
then $L_1 \cap L_2$ is reg.

contrapos: $L_1 \cap L_2$ not reg, at least one of L_1, L_2 is not reg.

know: $L_{0=1} \cap L(0^* 1^*)$ not regular.

know: $L(0^* 1^*)$ <u>is</u> regular





$\rightarrow$ $L_{0:1}$ is not regular.

---

more closure properties:

Lab $3\frac{1}{2}$ $\quad L$ is regular $\Rightarrow$ delete1(L) is regular

$$\text{delete1}(L) = \{ xy \mid x1y \in L \}$$

$$\{0^n 1^n \mid n \in \mathbb{N}\} = \text{delete1}\left(\boxed{\{0^n 1^{n+1} \mid n \in \mathbb{N}\}}\right).$$

$\uparrow$

not regular. $\longrightarrow$ not regular.

Some gotchas about idea of "remembering" infinitely many things vs "infinities" in the language.

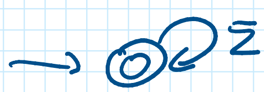
- idea: L_1, L_2 reg $\rightarrow L_1 \cap L_2$ reg

  used as $L_1 \cap L_2$ not reg $\rightarrow$ L_1 or L_2 is not reg.

confusing: seems like if we show $L' \subseteq L$ is nonregular, then L is non reg.

FALSE.

every L is a subset of Σ^*.

Σ^* is regular. 

consider $L = \{w \mid \#(0,w) \geq 374\}$. $F = \{0^n \mid n \in \mathbb{N}\}$? not fooling

— consider $L = \{ w \mid \#(0, w) \geq 374 \}$. $F = \{ 0^n \mid n \in \mathbb{N} \}$? ~~not fooling~~

0^{374} & 0^{375} are not distinguishable.

tempting to say ∞ values of $\#(0,w)$ to keep track of

turns out: keep track of $\#(0,w)$ in $\{0, \ldots, 374\}$.

once we read 374 th 0, we always accept.

→ big idea w/ non-regularity:

are there actually infinitely many configurations you need to distinguish? if so → Fooling set

if not → build DFA/NFA/regex.

bonus: Thm (Myhill-Nerode) that ties all of this together.