CSE322 Theory of Computation (L7)

Recap of last lecture

Questions of Tut4 = Exam standard
(expect you to solve in 15-20 mins.)

Quiz next lecture

Today

Closure Properties

Closure under Reverse()

$Reverse(L) = \{ rev(w) : w \in L \}$

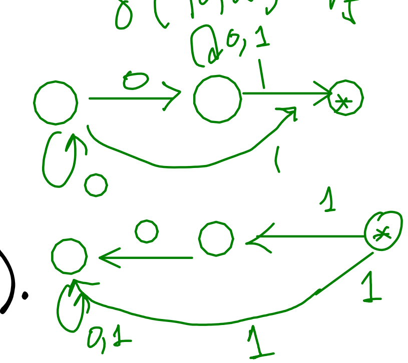
D: $q_0 \xrightarrow{w} q_f \in F$

$\Downarrow$

N: $q_0 \xleftarrow{rev(w)} q_f$

$q_1 \xleftarrow{rev(w)}$

$\hat{\delta}(q_0, w) = q_f$



Suppose D accepts L. Construct N that accepts $RL = Reverse(L)$.

$D = \langle Q, \Sigma, \delta, q_0, F \rangle$ $\quad$ $N = \langle Q1, \Sigma, \delta 1, q_{01}, F1 \rangle$

$q_{f1} \rightsquigarrow q_0$ reverse of $\hat{\delta}$

$q_s \xrightarrow{\varepsilon} q_{f1}$, $q_s \xrightarrow{\varepsilon} q_{f2}$, $q_s \xrightarrow{\varepsilon} \dots$

$Q1 = Q \cup \{q_s\}$ $\quad$ $F1 = \{q_0\}$

$q \xrightarrow{a} r$, $q \xleftarrow{a} r$

$q_{01} = \{q_s\}$ $\quad$ $\delta 1(q_s, \varepsilon) = F$, $\delta 1(q \neq q_s, a) = \{ r : \delta(r, a) = q \}$

Prove that: $\hat{\delta}(q_0, w) \in F$ iff $\hat{\delta 1}(q_{01}, rev(w)) \cap F1 \neq \emptyset$

D accepts w $\quad$ $rev(w) = wR$

$\hat{\delta}_1(q_s, wR) \ni q_0$

$q_s \xrightarrow{w}$

$\hat{\delta}_1(q_s, wR) = \hat{\delta}_1(q_s, wR \cdot \varepsilon) =$

$q_0 \in \bigcup_{r \in \delta 1(q_s, \varepsilon)} \hat{\delta}_1(r, wR) \iff \exists r \in \underbrace{\boxed{\delta 1(q_s, \varepsilon)}}_{F} \; q_0 \in \hat{\delta}_1(r, wR)$

Prove that: $\hat{\delta}(q_0, w) \in F$ iff $\exists r \in F$ s.t. $q_0 \in \hat{\delta}_1(r, wR)$

D: $q_0 \xrightarrow{w} q_f$ $\quad$ $q_0 \xleftarrow{wR} q_f$ N

$\Rightarrow$ Let $\hat{\delta}(q_0, w) \in F$. Show that $q_0 \in \hat{\delta}_1(q_f, wR)$. Follows from definition $\delta 1$ + induction

$= q_f$

$\Leftarrow$ Let $r \in F$ s.t. $q_0 \in \hat{\delta}_1(r, wR)$. Show that $\hat{\delta}(q_0, w) = r$

Closure under Half()

$L = \{1, 10, 100, 1100, 10101, 010101, \dots\}$

$Half(L) = \{1, \ , 11, \ , 010 \dots\}$

$Half(L) = \{ x :$ there is some y in L s.t. $|x| = |y|$ and $\underline{xy \text{ is in } L}\}$

$D = \langle Q, \Sigma, \delta, q_0, F \rangle$

$N = \langle Q1 = Q \times Q \cup \{q_s\}, \Sigma, \delta 1, q_s, F1 = \{(q,q) : q \in Q\} \rangle$

$\delta 1(q_s, \varepsilon) = \{(q_0, q_f) : q_f \in F\}$, $\delta 1(q_s, a) = \emptyset$

$\delta 1((q1, q2), a) = \{(\delta(q1, a), q') : \exists b, \delta(q', b) = q2\}$

D: $q_0 \xrightarrow{x} \bullet \xrightarrow{y} q_f \in F$

product automata

1. Which q_f? Guess
2. Which state should be a final state? Both end in the same state.
3. 2nd DFA should be simulated in reverse.
4. What should be y? Guess.

$\hat\delta(q_0, x) = q$ $\qquad$ $\hat\delta_1((q_0,q_f),x)$

Claim: for any x, $q_f \in F$: $\exists y \ \hat\delta(q_0, xy) = q_f$ iff $\cancel{\hat\delta_1(q_s, x)} \ni (q_f, q_f)$

Thm: $\exists y \ \hat\delta(q_0, xy) \in F$ iff $\hat\delta_1(q_s, x) \cap F1 \neq \emptyset$

$x \in Half(L)$ $\qquad$ N accepts x

For any $q_f \in F$

For any x, y

$\exists q$

iff

$q_0 \xrightarrow{x} q \xrightarrow{y} q_f \in F$ $\qquad$ $\hat\delta(q_0, x) = q$ & $\hat\delta(q, y) = q_f$

$\hat\delta_1((q_0, q_f), x) \ni (q, q)$

if $\exists b$, $q' \xrightarrow{b} q$
then $q \xrightarrow{a} q'$ $\Big\}$ guess of y

Guess.

$q_0 \xrightarrow{x} q$ $\qquad$ $q \xrightarrow{a} \delta(q, a)$

$q_f \xrightarrow{y_{rev}} q$ $\qquad$ $q \xrightarrow{a} q'$ if $\exists b \ \delta(q', b) = q$ $\qquad$ $q \xleftarrow{b} q'$

$q0 \xrightarrow{x} q_{mid} \xrightarrow{x} q_f \in F$

Closure under SameHalf()

$L = \{1, 00, 1010, 1001, \dots\}$

$\text{SameHalf}(L) = \{0, 10, \dots\}$

$\text{SameHalf}(L) = \{ x : xx \text{ is in } L \} \longrightarrow N = \langle Q1 = \{q_s\} \cup \underbrace{Q}_{\text{1st half}} \times \underbrace{Q}_{\text{store guess}} \times \underbrace{Q}_{\text{2nd half}}, \Sigma, \delta 1, q_s, F1 = \cancel{Q \times Q \times F} \rangle$

$\{(q, q, q_f) : q \in Q, q_f \in F\}$

$D = \langle Q, \Sigma, \delta, q0, F \rangle$

① Simulate D on x → 1st half

② Parallely simulate D on x for the second half → how to choose the middle state → guess

③ Product DFA is successful when 1st half ends at the guessed state
2nd " " $q_f \in F$

$\delta_1(q_s, \varepsilon) = \{(q0, r, r) : r \in Q\}$

$\delta_1(q_s, a) = \emptyset$

$\delta_1((q1, q2, q3), a) = \{(\delta(q1, a), q2, \delta(q2, a))\}$

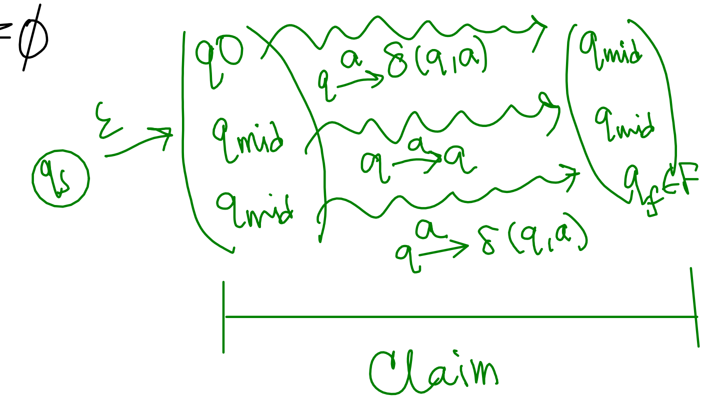
no change

$\hat{\delta}_1((q_0, q_m, q_m), x) \ni (q_m, q_m, q_f)$

Claim : $\hat{\delta}(q0, xx) = q_f$ and $\hat{\delta}(q0, x) = q_m$ iff $\hat{\delta}_1(q_s, x) \ni (q_m, q_m, q_f)$

Thm : $\hat{\delta}(q0, xx) \in F$ iff $\hat{\delta}_1(q_s, x) \cap F1 \neq \emptyset$

$x \in \text{SameHalf}(L)$ NFA accepts x



Pumping Lemma

If L is a regular language, then ...

Does not say the value of p

There exists a positive p s.t.

For all strings w in L of length p or more,

pumpable strings

There exists a partitioning w = xyz, where

1. xz is in L, xyz is in L, xyyz is in L, xyyyz is in L, xyyyyz is in L, xyyyyyz is in L, ...
2. |xy| <= p
3. y is not empty string (x,z could be empty)

* PL does not say what is the pumpable subseq.

Proof of Pumping Lemma

Proof by construction.

L is recognized by DFA M with s states. Set p=s.

w is a 'long' string i.e., |w| >= s.
in L

if $q3 \notin F$, $xz, xyz, xyyz \cdots \notin L$

states after each symbol

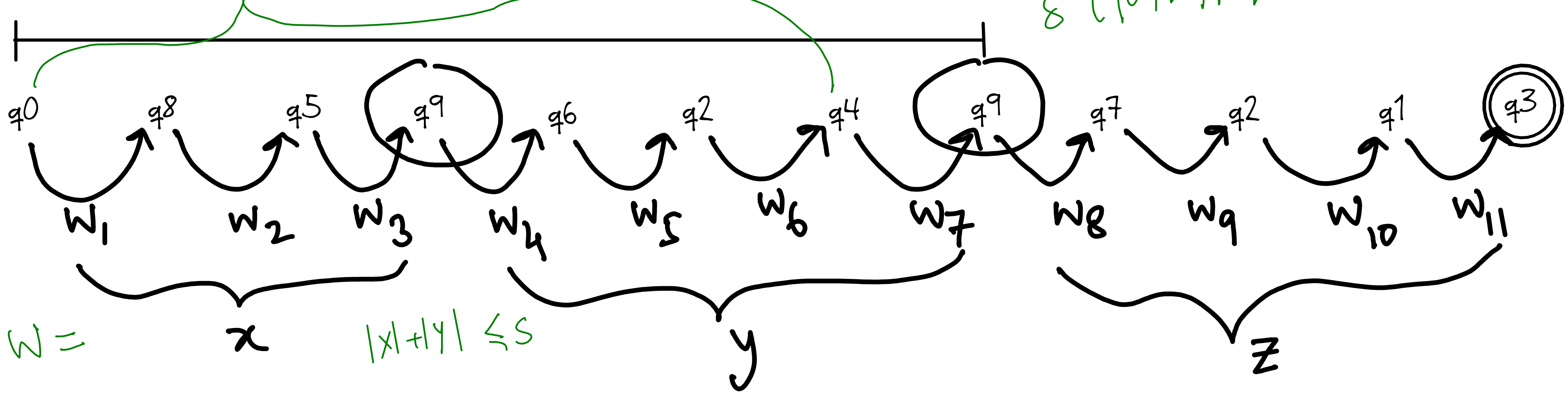
$q0 \rightarrow q8 \rightarrow q5 \rightarrow q9 \rightarrow q6 \rightarrow q2 \rightarrow q4 \rightarrow q9 \rightarrow q7 \rightarrow q2 \rightarrow q1 \rightarrow q3$

states in above sequence ? $|w|+1 \geq |s|+1$

$\hat{\delta}(q0, xz) = q3$

$\hat{\delta}(q0, xyyz) = q0 \overset{x}{\rightsquigarrow} q9 \overset{y}{\rightsquigarrow} q9 \overset{y}{\rightsquigarrow} q9 \overset{z}{\rightsquigarrow} q3$

states upto first repeating state? $\leq s$



W =

$|x|+|y| \leq s$

Pumping Lemma

If L is a regular language, then ...

Does not say the value of p

There exists a positive p s.t.

For all strings w ~~in L~~ of length p or more, pumpable strings

There exists a partitioning w = xyz, where

1. xz is in L, xyz is in L, xyyz is in L, xyyyz is in L, xyyyyz is in L, xyyyyyz is in L, ... (or, all NOT in L)
2. |xy| <= p
3. y is not empty string (x,z could be empty)

* PL does not say what is the pumpable subseq.

Proof of Non-regular Languages

Given L, how to prove that L is not regular ...

That is, L CANNOT be accepted by a DFA?

Proof by contradiction: Assume that L is regular. PL applies to L.
For any pumpable w, PL claims must hold.

Proof: Use Pumping Lemma <u>to construct</u>

a pumpable w in L (or w not in L) s.t. w can be

pumped (up/down) to get w' NOT in L (or w' in L).

Take p assured by PL. Take some cleverly chosen w of large length.
Arrive at a contradiction by showing that some pumped versions of w
doesn't agree with w. (Try all possible partitioning.)

$L = \{0^n 1^n \mid n >= 0\}$ is not regular

$\{\varepsilon, 01, 0011, 000111, \ldots\}$

Proof by contradiction, using Pumping Lemma.

PL guarantees p s.t. for every string w of length >= p, there is a non-empty subsequence which can be pumped.

$|xy| \le p$

Select $w = 0^p 1^p$  $\#(0,w) < \#(1,w)$.

$\in L$.

$$\overbrace{0\;0\;\underbrace{0\ldots0}_{y}\;0\;0}^{p}\;1\;1\;1\ldots1\;1\;1 \implies$$

y consists of only 0's.

$y \neq \varepsilon$

xz has more 1's than 0's

$\therefore xz \notin L$.

$0^{p/2} 1^{p/2}$.

Q: Can we use w = 0...(p/2 times)...01...(p/2 times)...1