Intro to Theory of Computation

Lecture 5

Last time:
• Closure properties.
• Equivalence of NFAs, DFAs and regular expressions

Today:
• Conversion from NFAs to regular expressions
• Proving that a language is not regular: pumping lemma

Sofya Raskhodnikova

Sofya Raskhodnikova; based on slides by Nick Hopper
Theorem. Every NFA has an equivalent regular expression.

Proof idea: Transform NFA to a regular expression by removing states and relabeling the arrows with regular expressions.
What is the regular expression that generates all strings that take this machine from $q_s$ to $q_a$?

A. $ab^*c \cup d$
B. $ab^*c \cap d$
C. $abc \cup d$
D. $(ab^*cd)^*$
E. None of the above.
Generalized NFAs

- Each transition is labeled with a regular expression
- Unique and distinct start and accept states
- No transitions to the start state
- No transitions from the accept state
Generalized NFAs

G accepts w if it finds $q_0 q_1 \ldots q_k, \ w_1 \ldots w_k$:
- $R(q_s, q) = a^* b$
- $R(q_i, q) = \emptyset$
- $w = w_1 w_2 \ldots w_k$
- $R(q, q_k) \neq \emptyset$
- $q_0 = q_s, \ q_k = q_a$
NFA to GNFA

Add a new start state with no incoming arrows. Make a unique accept state with no outgoing arrows.
While machine has more than 2 states:

Pick an internal state, rip it out and relabel the arrows with regular expressions to account for the missing state.
$R(q_0, q_3) = (a^*b)(a \lor b)^*$
bb ∪ (abb ba)b*a = R(q₁,q₁)
\(bb \cup (a \cup ba)b^*a = R(q_1, q_1)\)

\(b \cup (a \cup ba)b^* = R(q_1, q_a)\)

\((bb \cup (a \cup ba)b^*a)^*(b \cup (a \cup ba)b^*)\)
Convert the NFA to a regular expression

\[ \varepsilon \cup (a \cup b)b^*b(bb^*b)^* \]
Convert the NFA to a regular expression

\[ \varepsilon \cup (a \cup b)b*b(bb*b)*a \]

\[ (((a \cup b)b*b(bb*b)*a)\varepsilon \cup (a \cup b)b*b(bb*b)* \]
Add $q_{\text{start}}$ and $q_{\text{accept}}$ to create GNFA $G$. 
Run CONVERT($G$)

CONVERT($G$):
If $\#\text{states} \geq 2$

return the expression on the arrow going from $q_{\text{start}}$ to $q_{\text{accept}}$
NFA to regular expression

Add $q_{\text{start}}$ and $q_{\text{accept}}$ to create GNFA $G$.

Run CONVERT($G$)

CONVERT($G$):

If #states $> 2$

Build $G'$ from $G$:

- select $q_{\text{rip}} \in Q$ different from $q_{\text{start}}$ and $q_{\text{accept}}$
- define $Q' = Q - \{q_{\text{rip}}\}$
- define $R'$ as:
  $$R'(q_i, q_j) = R(q_i, q_{\text{rip}})R(q_{\text{rip}}, q_{\text{rip}})^*R(q_{\text{rip}}, q_j) \cup R(q_i, q_j)$$

return CONVERT($G'$)
Conversion procedures

DFA ↔ NFA

definition

Regular Language

Regular Expression

Sofya Raskhodnikova; based on slides by Nick Hopper
Design an NFA for the language:

\[ \{0^n1^n \mid 0 < n \leq 2\} \]

\[ \{0^n1^n \mid 0 < n \leq k\} \]

\[ \{0^n1^n \mid n > 0\}? \]

(For R a regexp, \( R^2 \) means \( RR \), and \( R^n \) means \( RR \ldots R \))
SOME LANGUAGES ARE NOT REGULAR!

\[ B = \{0^n1^n \mid n \geq 0\} \] is NOT regular!
Proof (by contradiction)

Let M be a k-state DFA that recognizes B.

Consider the path M takes on $0^k1^k$:

$q_0q_1q_2\ldots q_iq_{i+1}q_jq_k\ldots q_{2k} \in F$

0000\ldots00..0..011111\ldots11

There must be $i < j \leq k$ such that $q_i = q_j$

M accepts $0^{k-(j-i)}1^k \notin B!$

So M does not recognize the language B.
REGULAR OR NOT?

C = \{ w \mid w \text{ has equal number of } 1\text{s and } 0\text{s}\}

NOT REGULAR

D = \{ w \mid w \text{ has equal number of occurrences of } 01 \text{ and } 10\} \\
(0\Sigma^*0) \cup (1\Sigma^*1) \cup 1 \cup 0 \cup \varepsilon
THE PUMPING LEMMA

Let $L$ be a regular language with $|L| = \infty$

Then there exists a length $p$ such that

if $w \in L$ and $|w| \geq p$ then

$w$ can be split into three parts $w=xyz$ where:

1. $|y| > 0$
2. $|xy| \leq p$
3. $xy^iz \in L$ for all $i \geq 0$
THE PUMPING LEMMA

Example:

Let \( L = 0^*1^*; p = 1 \)
\( w = 011 \)
\( x = \varepsilon \)
\( y = 0 \)
\( z = 11 \)

if \( w \in L \) and \( |w| \geq p \) then \( w = xyz \), where:

1. \( |y| > 0 \)

Let \( L = (0 \cup 1)^2^*; p = 2 \)
\( w = 12 \)
\( x = 1 \)
\( y = 2 \)
\( z = \varepsilon \)

Sofya Raskhodnikova; based on slides by Nick Hopper
Let M be a DFA that recognizes L.
Let $p$ be the number of states in M.
Assume $w \in L$ is such that $|w| \geq p$.

We show $w = xyz$

Proof of the pumping lemma

1. $|y| > 0$
2. $|xy| \leq p$
3. $xy^iz \in L$ for all $i \geq 0$

There must be $j > i$ such that $q_i = q_j$. 
Use the pumping lemma to prove that $B = \{0^n1^n \mid n \geq 0\}$ is not regular

**Hint:** Assume $B$ is regular, and try pumping $w = 0^p1^p$

If $B$ is regular, $w$ can be split into $w = xyz$, where

1. $|y| > 0$
2. $|xy| \leq p$
3. $xy^iz \in B$ for all $i \geq 0$

$y$ is all 0s: $xzyz$ has more 0s than 1s

**Contradiction!**
Proof by contradiction: assume $L$ is regular.

Then there is a pumping length $p$.

Find a string $w \in L$ with $|w| \geq p$.

Show that no matter how you choose $xyz$, $w$ cannot be pumped!

Conclude that $L$ is not regular.
PALINDROMES = \{ \text{ww}^R \mid w \in \{0,1\}^* \} \text{ is not regular.}

Proof: Assume ... pumping length \( p \)

Find a \( w \in \text{PALINDROMES} \) longer than \( p \)

\[ 0^p1^p1^p0^p \]

Show that \( w \) cannot be pumped:

\[
\begin{align*}
\text{w} &= \underbrace{00 \ldots 00}_{p} \underbrace{11 \ldots 11}_{2p} \underbrace{00 \ldots 00}_{p} \\
\text{xyyz} &= \underbrace{00 \ldots 00}_{> p} \underbrace{00011 \ldots 1100 \ldots 00}_{2p} \underbrace{00 \ldots 00}_{p}
\end{align*}
\]

\( y \) must be in this part
PALINDROMES = \{ ww^R \mid w \in \{0,1\}^* \} is not regular.

Proof: Assume ... pumping length p

Find a \( w \in \text{PALINDROMES} \) longer than \( p \)

\[ 0^p1^p1^p0^p \]

Show that \( w \) cannot be pumped:

If \( w = xyz \) with \( |xy| \leq p \) then
\( y = 0^J \) for some \( J > 0 \).

Then \( xyyz = 0^{p+J}1^{2p}0^p \notin \text{PALINDROMES} \)

Contradiction!
Prove $C = \{ 0^i1^j \mid i > j \geq 0 \}$ is not regular.

Proof: Assume ... pumping length $p$

Find a $w \in C$ longer than $p$

$0^{p+1}1^{p}$

Show that $w$ cannot be pumped:

$w = 00...0011...11$

$y$ must be in this part

$xyz = 00...00011...11$  $xz = 0...0011...11$

$> p+1$  $\leq p$
Prove $C = \{ 0^i1^j \mid i > j \geq 0 \}$ is not regular.

Proof: Assume ... pumping length $p$

Find a $w \in C$ longer than $p$

$0^{p+1}1^p$

If $w = xyz$ with $|xy| \geq p$ then $y = 0^J$ for some $J \geq 1$.

Then $xy^0z = xz = 0^{p+1-J}1^p \notin C$

Contradiction!
1. YOU pick the language $L$ to be proved nonregular.

2. ADVERSARY picks $p$, but doesn't reveal to YOU what $p$ is; YOU must devise a play for all possible $p$'s.

3. YOU pick $w$, which should depend on $p$ and which must be of length at least $p$.

4. ADVERSARY divides $w$ into $x, y, z$, obeying conditions stipulated in the pumping lemma: $|y| > 0$ and $|xy| \leq n$. Again, ADVERSARY does not tell YOU what $x, y, z$ are, although they must obey the constraints.

5. YOU win by picking $i$, which may be a function of $p, x, y, z$, such that $xy^iz$ is not in $L$. 

Sofya Raskhodnikova; based on slides by Nick Hopper