### Outline

- Nondeterministic Finite Automata
- Closure under regular operations
- NFA, FA equivalence
- Regular Expressions
- FA and Regular Expressions

---

#### NFA diagrams

- **States**
  - (single) start state
  - (several) accept states
- **Transitions**:
  - may have several with a given label (or none)
  - may be labeled with $\varepsilon$
- **At each step, several choices for next state**

---

#### NFA operation

- **Example of NFA operation:**
  - alphabet $\Sigma = \{0, 1\}$
  - input: $0 1 0$
  - not accepted

---

#### NFA operation

- **Example of NFA operation:**
  - alphabet $\Sigma = \{0, 1\}$
  - input: $1 1 0$
  - accepted

---

#### NFA operation

- **One way to think of NFA operation:**
  - string $x = x_1 x_2 x_3 \ldots x_n$ accepted if and only if
    - there exists a way of inserting $\varepsilon$'s into $x$
    - so that there exists a path of transitions from the start state to an accept state
A nondeterministic FA is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$ -
- $Q$ is a finite set called the states
- $\Sigma$ is a finite set called the alphabet
- $\delta: Q \times (\Sigma \cup \{\epsilon\}) \rightarrow \mathcal{P}(Q)$ is a function called the transition function
- $q_0$ is an element of $Q$ called the start state
- $F$ is a subset of $Q$ called the accept states

Formal description of NFA operation

NFA $M = (Q, \Sigma, \delta, q_0, F)$
accepts a string $w = w_1 w_2 w_3 \ldots w_n \in \Sigma^*$
if $w$ can be written (by inserting $\epsilon$'s) as:
$y = y_1 y_2 y_3 \ldots y_m \in (\Sigma \cup \{\epsilon\})^*$
and $\exists$ sequence $r_0, r_1, \ldots, r_m$ of states for which
- $r_0 = q_0$
- $r_{i+1} \in \delta(r_i, y_{i+1})$ for $i = 0, 1, 2, \ldots, m-1$
- $r_m \in F$

Closures

- Recall: want to show the set of languages recognized by NFA is closed under:
  - union $^*C = (A \cup B)^*$
  - concatenation $^*C = (A \circ B)^*$
  - star $^*C = A^*$

Closure under union

$C = (A \cup B) = \{ x : x \in A \text{ or } x \in B \}$

Closure under concatenation

$C = (A \circ B) = \{ xy : x \in A \text{ and } y \in B \}$
Closure under star

\[ C = A^* = \{ x_1x_2x_3\ldots x_k : k \geq 0 \text{ and each } x_i \in A \} \]

NFA, FA equivalence

**Theorem:** A language \( L \) is recognized by a FA if and only if \( L \) is recognized by a NFA.

Must prove two directions:

\((\Rightarrow)\) \( L \) is recognized by a FA implies \( L \) is recognized by a NFA.

\((\Leftarrow)\) \( L \) is recognized by a NFA implies \( L \) is recognized by a FA.

(usually one is easy, the other more difficult)

NFA, FA equivalence

\((\Rightarrow)\) \( L \) is recognized by a FA implies \( L \) is recognized by a NFA.

**Proof:** A finite automaton is a nondeterministic finite automaton that happens to have no \( \varepsilon \)-transitions, and for which each state has exactly one outgoing transition for each symbol.

NFA, FA equivalence

\((\Leftarrow)\) \( L \) is recognized by a NFA implies \( L \) is recognized by a FA.

**Proof:** We will build a FA that simulates the NFA (and thus recognizes the same language).

- alphabet will be the same
- what are the states of the FA?

NFA, FA equivalence

- given NFA \( M = (Q, \Sigma, \delta, q_0, F) \)
- construct FA \( M' = (Q', \Sigma', \delta', q_0', F') \)
- same alphabet: \( \Sigma' = \Sigma \)
- states are subsets of \( M \)'s states: \( Q' = \mathcal{P}(Q) \)
- if we are in state \( R \in Q' \) and we read symbol \( a \in \Sigma' \), what is the new state?

NFA, FA equivalence

- given NFA \( M = (Q, \Sigma, \delta, q_0, F) \)
- construct FA \( M' = (Q', \Sigma', \delta', q_0', F') \)
- **Helpful def'n:** \( E(S) = \{ q \in Q : q \text{ reachable from } S \text{ by traveling along 0 or more } \varepsilon \text{-transitions} \} \)
- new transition fn: \( \delta'(R, a) = \bigcup_{r \in R} E(\delta(r, a)) \)
  = "all nodes reachable from \( R \) by following an \( a \)-transition, and then 0 or more \( \varepsilon \)-transitions"
NFA, FA equivalence

- given NFA $M = (Q, \Sigma, \delta, q_0, F)$
- construct FA $M' = (Q', \Sigma', \delta', q'_0, F')$

  - new start state: $q'_0 = E(q_0)$
  - new accept states:
    $F' = \{ R \in Q' : R$ contains an accept state of $M \}$

We have proved ($\Leftarrow$) by construction.

Formally we should also prove that the construction works, by induction on the number of steps of the computation.

- at each step, the state of the FA $M'$ is exactly the set of reachable states of the NFA $M$.

So far...

**Theorem**: the set of languages recognized by NFA is closed under union, concatenation, and star.

**Theorem**: a language $L$ is recognized by a FA if and only if $L$ is recognized by a NFA.

**Theorem**: the set of languages recognized by FA is closed under union, concatenation, and star.

Next...

- Describe the set of languages that can be built up from:
  - unions
  - concatenations
  - star operations
- Called “patterns” or regular expressions
- **Theorem**: a language $L$ is recognized by a FA if and only if $L$ is described by a regular expression.

Regular expressions

- $R$ is a regular expression if $R$ is
  - $a$, for some $a \in \Sigma$
  - $\varepsilon$, the empty string
  - $\emptyset$, the empty set
  - $(R_1 \cup R_2)$, where $R_1$ and $R_2$ are reg. exprs.
  - $(R_1 \circ R_2)$, where $R_1$ and $R_2$ are reg. exprs.
  - $(R_1^*)$, where $R_1$ is a regular expression

A reg. expression $R$ describes the language $L(R)$.

- example: $R = (0 \cup 1)$
  - if $\Sigma = \{0,1\}$ then use “$\Sigma$” as shorthand for $R$
- example: $R = 0 \circ \Sigma^*$
  - shorthand: omit “$\circ$” $R = 0\Sigma^*$
  - precedence: $\circ$, then $\cup$, unless override by parentheses
  - in example $R = 0(\Sigma^*)$, not $R = (0\Sigma)^*$
Some examples

alphabet
\( \Sigma = \{0,1\} \)

- \{w : w has at least one 1\}
  \[= \Sigma^*1\Sigma^*\]

- \{w : w starts and ends with same symbol\}
  \[= 0\Sigma^*0 \cup 1\Sigma^*1\cup 0 \cup 1\]

- \{w : |w| \leq 5\}
  \[= (\varepsilon \cup \Sigma)(\varepsilon \cup \Sigma)(\varepsilon \cup \Sigma)(\varepsilon \cup \Sigma)(\varepsilon \cup \Sigma)\]

- \{w : every 3\text{rd} position of w is 1\}
  \[= (1\Sigma^*)^*(\varepsilon \cup 1 \cup 1\Sigma)\]