Nondeterministic Finite Automata

Lecture 6
Section 2.2

Robb T. Koether

Hampden-Sydney College

Mon, Sep 5, 2016

Outline

Outline

Determinism

- A deterministic finite automaton is deterministic because every move is forced.
- That is, δ is a function.
- For every state-symbol combination (q, x) in $Q \times \Sigma$, there is *exactly one* $q' \in Q$ such that $\delta(q, x) = q'$.

Nondeterminism

- To make a finite automaton *nondeterministic*, we drop the requirement that the image of (q, x) be a unique state and allow it to be a set of states.
- (q, x) may have no image, one image, or more than one image.

Nondeterminism

Definition (λ-move)

An λ-move is a transition from one state to another made without reading an input symbol. (We "read" λ.)

- We also allow "λ-moves."

Outline

1. Nondeterminism
   - Definition
   - Examples
2. Building a DFA from an NFA
3. Assignment

Definition of a Nondeterministic Finite Automaton

Definition (Nondeterministic finite automaton)

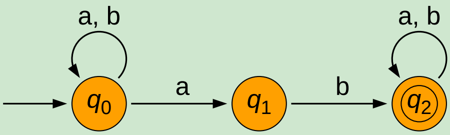
A nondeterministic finite automaton (NFA) is a 5-tuple $\{Q, \Sigma, \delta, q_0, F\}$, where

- Q, Σ, q_0, and F are as they were for a DFA.
- The transition function is

$$\delta : Q \times (\Sigma \cup \{\lambda\}) \to \mathcal{P}(Q).$$

Example

Example (NFA)



The Language of an NFA

Definition (Acceptance by an NFA)

A string *w* is accepted by an NFA if there is *at least one* computation on the NFA with input *w* that terminates in an accepting state.

Definition (Language of an NFA)

The language of an NFA is the set of all strings in Σ^* that are accepted by the NFA.

- For a given input, an NFA may admit a multitude of computations.

Outline

Example (Nondeterministic finite automata)

- Let $\Sigma = \{\mathbf{a}, \mathbf{b}\}$.
- Let $L_1 = \{w \in \Sigma^* \mid w$ contains an even number of **a**'s$\}$.
- Let $L_2 = \{w \in \Sigma^* \mid w$ contains an even number of **b**'s$\}$.
- Design NFAs that accept
  - $L_1 \cup L_2$
  - $L_1 L_2$
  - L_1^*
  - $L_1 \cap L_2$
- Describe δ for the NFA that accepts AB.
- Do the computation for the strings **ababb** and **ababbb**.

Outline

Relations and Functions

- Given a function

$$f\colon A \to \mathcal{P}(B),$$

we may derive a function

$$g\colon \mathcal{P}(A) \to \mathcal{P}(B).$$

Example

Example

- Let $A = \{2, 3, 4, 5\}$.
- Let $B = \{6, 7, 8, 9\}$.
- Let f be the function that maps every integer in A to the set of its multiples in B.

$$
\begin{aligned}
f &: A \to \mathcal{P}(B) \\
f(2) &= \{6, 8\}, \\
f(3) &= \{6, 9\}, \\
f(4) &= \{8\}, \\
f(5) &= \varnothing.
\end{aligned}
$$

Deriving $g : \mathcal{P}(A) \to \mathcal{P}(B)$

- Define $g : \mathcal{P}(A) \to \mathcal{P}(B)$ as follows.
- For each subset $S \subseteq A$, define

$$g(S) = \bigcup_{a \in S} f(a).$$

Example

Example (Deriving $g : \mathcal{P}(A) \to \mathcal{P}(B)$)

- Then, for example,

$$\begin{aligned} g(\{2\}) &= f(2) = \{6, 8\}, \\ g(\{2, 3\}) &= f(2) \cup f(3) = \{6, 8, 9\}, \\ g(\{2, 3, 4\}) &= f(2) \cup f(3) \cup f(4) = \{6, 8, 9\}. \end{aligned}$$

- What is $g(\varnothing)$?
- What is $g(\{4, 5\})$?

Outline

Collected

To be collected on Wed, Sep 7:

- Section 1.1 Exercises 31, 43a.
- Section 1.2 Exercises 15, 17e.
- Section 2.1 Exercises 7e, 17, 22.

Assignment

Assignment

- Section 2.2 Exercises 3, 5, 7, 8, 9, 12, 13, 14.