Theory of Computation Slides
based on Michael Sipser's Textbook

Moses A. Boudourides[1]

Visiting Associate Professor of Computer Science
Haverford College

[1] Moses.Boudourides@cs.haverford.edu

Section 1.2

Nondeterminism

January 25, 2022

Definition of Nondeterministic Finite Automata (NFA), I

Definition: A Nondeterministic Finite Automaton

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

- Q is a finite set called the **states**,
- Σ is a finite set called the **alphabet**,
- $q_0 \in Q$ is the **start state**,
- $F \subseteq Q$ is the **set of accepting** or (**final**) **states**, and
- $\delta \colon Q \times (\Sigma \cup \{\varepsilon\}) \to \mathcal{P}(Q)$ is the **transition function**, where $\mathcal{P}(Q)$ denotes the **power set** of the set of states Q.

For any element q of Q and any symbol $\sigma \in \Sigma$, we interpret $\delta(q, \sigma)$ as the set of states to which the NFA moves, when it is in state q and receives the input σ, or, if $\sigma = \varepsilon$, the set of states other than q to which the NFA can move from state q without receiving any input symbol.

Definition of Nondeterministic Finite Automata (NFA), II

Remarks on the Definition of NFA

- **Nondeterminism**: a string may follow more than one paths.
- ε**–moves**: a state can follow from a different state without receiving (reading) any input.
- **Empty states**: since the range of the transition function is the power set, a pair (q, σ) may be mapped to $\varnothing$ and, thus, there would be no arrow moving out of q labeled with σ.

Definition: ε–**Closures of States**

For any state p of a NFA, the ε–**closure** of p is defined to be a set denoted as $\varepsilon(p)$ consisting of all states q such that there is a path of arrows from p to q such that all the arrows of this path are labeled with ε. Notice that always $p \in \varepsilon(p)$ is true.

Equivalence of NFAs and DFAs

Theorem: **Equivalence of NFAs and DFAs**

Two FA are said **equivalent** if they recognize the same language.

Every NFA corresponds to an equivalent DFA.

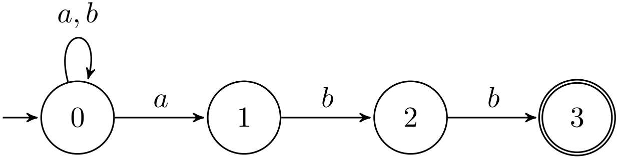
Algorithm for the Reduction of NFA to DFA

- Construct a table with the following columns:

q	$\delta(q,a)$	$\delta(q,b)$	$\delta(q,\varepsilon)$	$\varepsilon(q)$	$\delta(\varepsilon(q),a)$	$\delta(\varepsilon(q),b)$	$\varepsilon(\delta(\varepsilon(q),a))$	$\varepsilon(\delta(\varepsilon(q),b))$

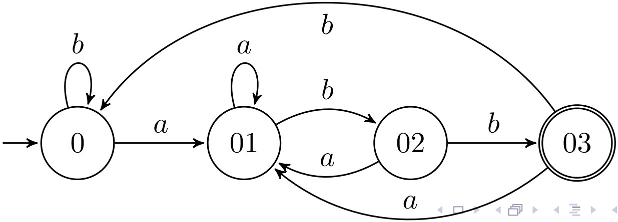
- If the NFA is without ε–moves, use only the first three columns!
- Concatenating notation of states:
  - Use subscripts of states!
  - $\{q_i\}\cup\{q_j\}\cup\{q_k\}=q_i+q_j+q_k=$ (denoted as) ijk (etc.)
- Underline final states in the table.
- Draw the equivalent DFA from the last two columns of the table.

Example: Reduction of NFA without ε–moves to DFA

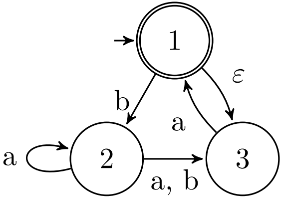


$$L(M) = \{x \in (a+b)^* \mid x \text{ ends in } abb\}$$
$$= (a+b)^* abb.$$

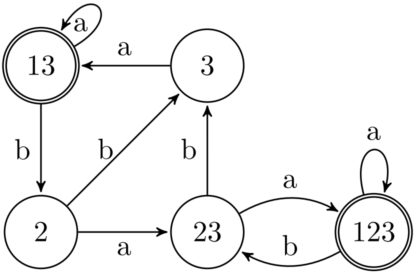
q	$\delta(q,a)$	$\delta(q,b)$
0	01	0
1	$\varnothing$	2
2	$\varnothing$	$\underline{3}$
3	$\varnothing$	$\varnothing$



Example: Reduction of NFA with ε–moves to DFA

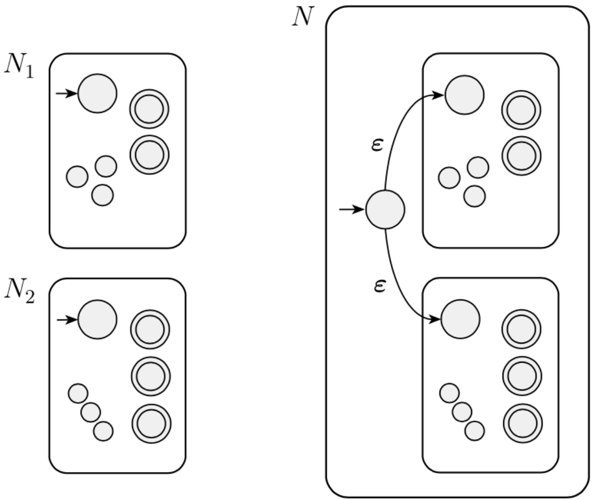


q	$\delta(q,a)$	$\delta(q,b)$	$\delta(q,\varepsilon)$	$\varepsilon(q)$	$\delta(\varepsilon(q),a)$	$\delta(\varepsilon(q),b)$	$\varepsilon(\delta(\varepsilon(q),a))$	$\varepsilon(\delta(\varepsilon(q),b))$
1	$\varnothing$	2	3	13	1	2	$\underline{13}$	2
2	23	3	$\varnothing$	2	23	3	23	3
3	1	$\varnothing$	$\varnothing$	3	1	$\varnothing$	$\underline{13}$	$\varnothing$



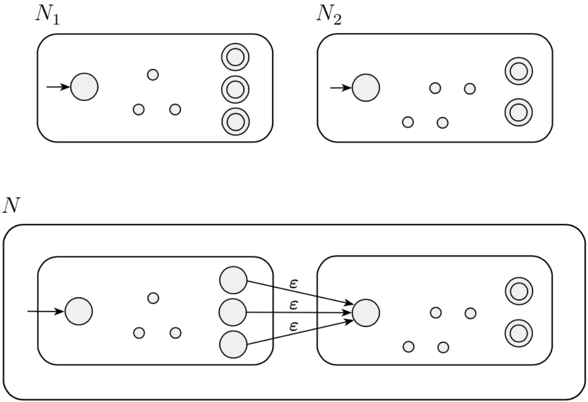
Union of two FAs

If $L_1 = L(N_1)$ and $L_2 = L(N_2)$, then $L_1 \cup L_2$ is recognized (accepted) by the NFA N, i.e., $L_1 \cup L_2 = L(N)$.



Concatenation of two FAs

If $L_1 = L(N_1)$ and $L_2 = L(N_2)$, then $L_1 \circ L_2$ is recognized (accepted) by the NFA N, i.e., $L_1 \circ L_2 = L(N)$.



The Star of a FA

If $L_1 = L(N_1)$, then L_1^* is recognized (accepted) by the NFA N, i.e., $L_1^* = L(N)$.