Formal Languages

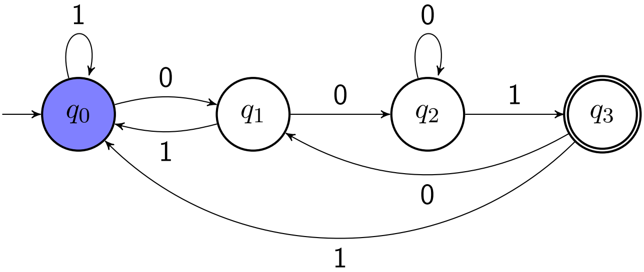
Philippe de Groote

2020-2021

Outline

2. Finite state automata
   - Introduction
   - Deterministic finite state automata
   - Non-deterministic finite state automata
   - Equivalence of NFSA and DFSA
   - Finite state automata with ϵ-transitions
   - Elimination of ϵ-transitions

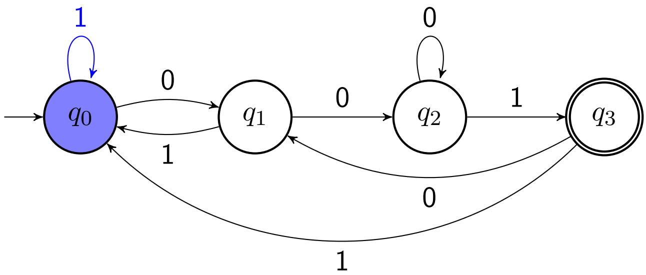
Introduction



1 0 1 0 0 1 0 0 0 1

↑

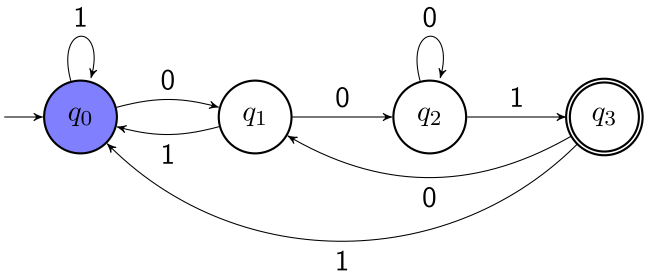
Introduction



1 0 1 0 0 1 0 0 0 1

↑

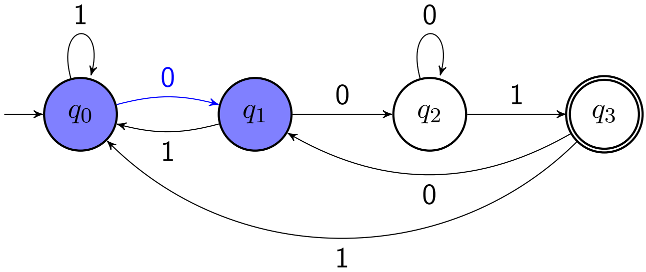
Introduction



1 0 1 0 0 1 0 0 0 1

↑

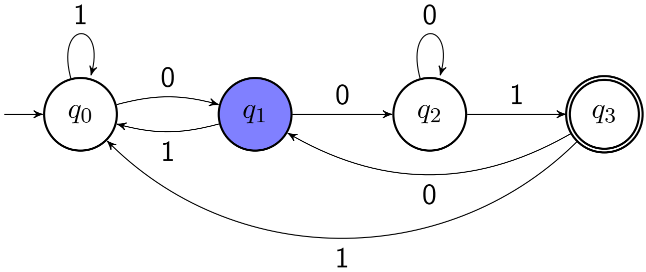
Introduction



1 0 1 0 0 1 0 0 0 1

↑

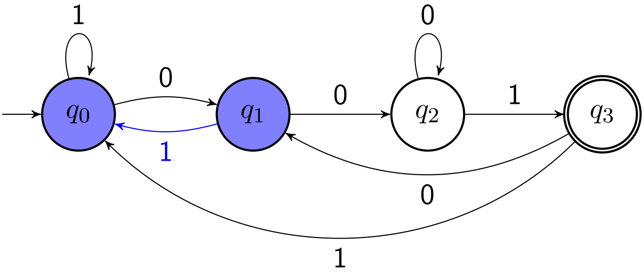
Introduction



$1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1$

$\uparrow$

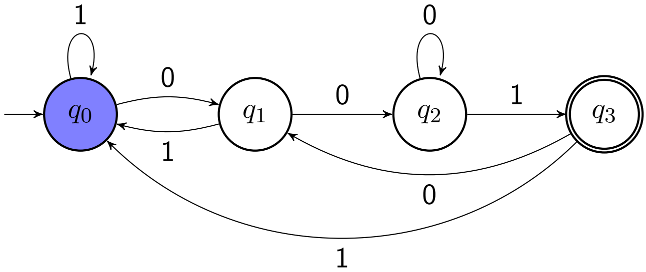
Introduction



$1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1$

$\uparrow$

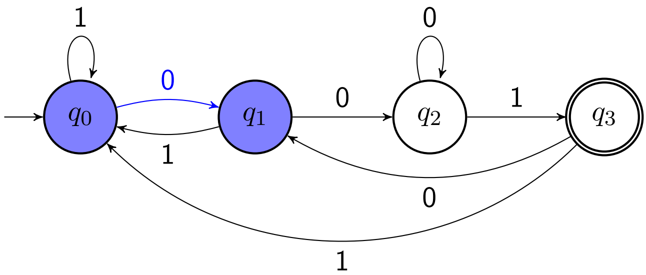
Introduction



1 0 1 0 0 1 0 0 0 1

↑

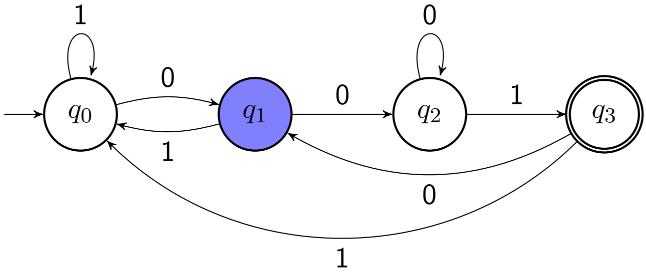
Introduction



1 0 1 0 0 1 0 0 0 1

↑

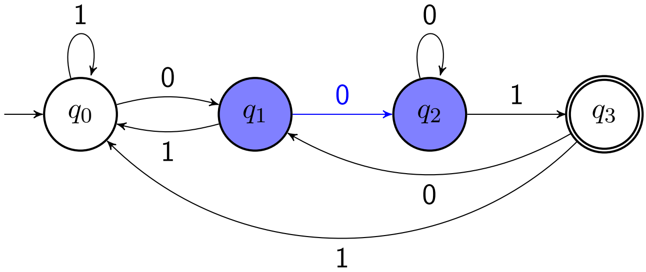
Introduction



$1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1$

$\uparrow$

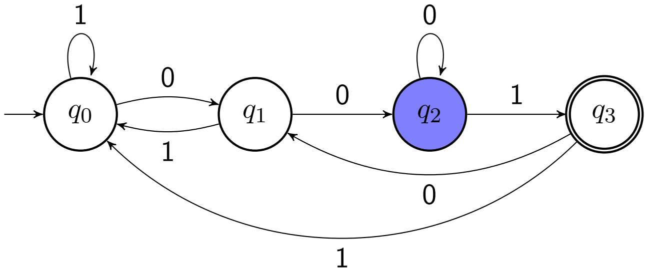
Introduction



1 0 1 0 0 1 0 0 0 1

↑

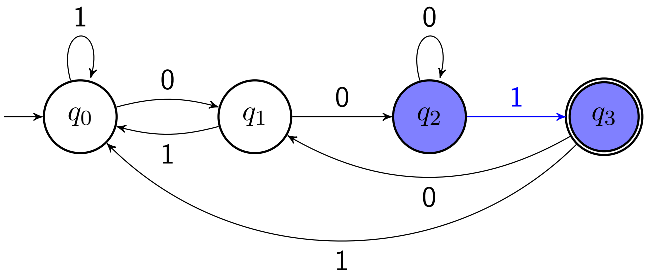
Introduction



1 0 1 0 0 1 0 0 0 1

↑

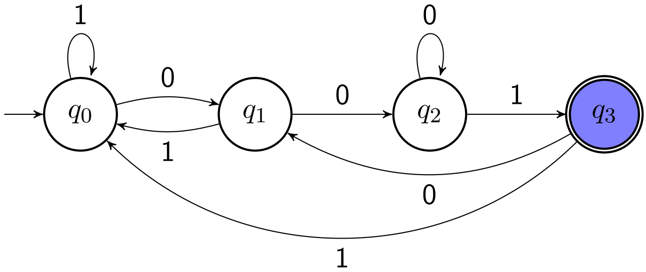
Introduction



1 0 1 0 0 1 0 0 0 1

↑

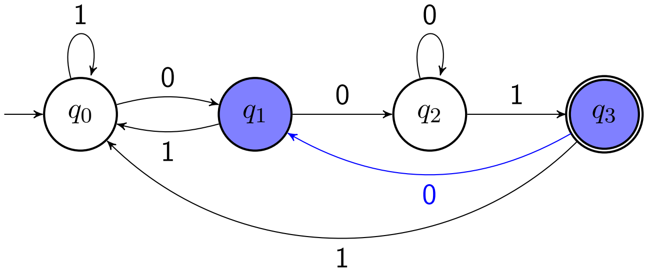
Introduction



1 0 1 0 0 1 0 0 0 1

↑

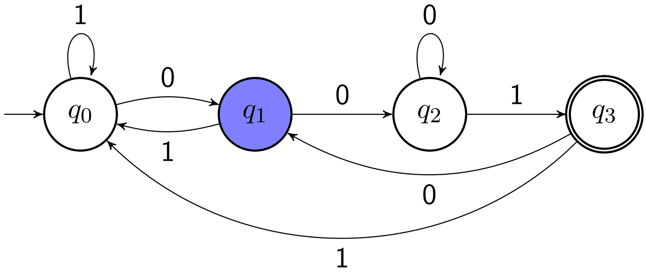
Introduction



1 0 1 0 0 1 0 0 0 1

↑

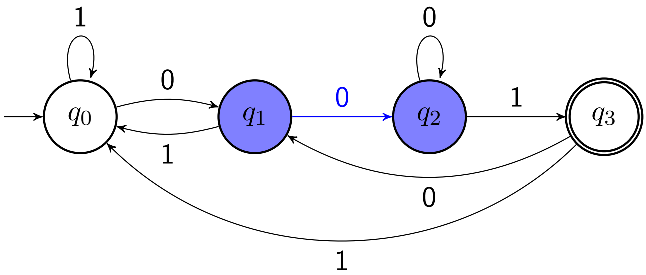
Introduction



$1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1$

$\uparrow$

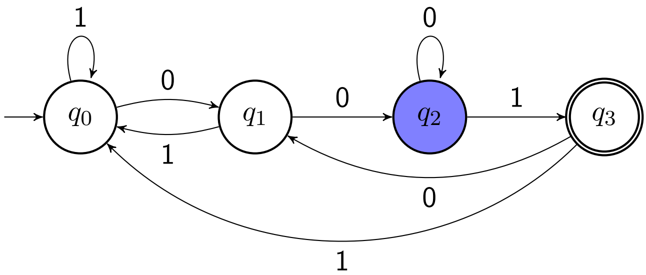
Introduction



$1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 1$

$\uparrow$

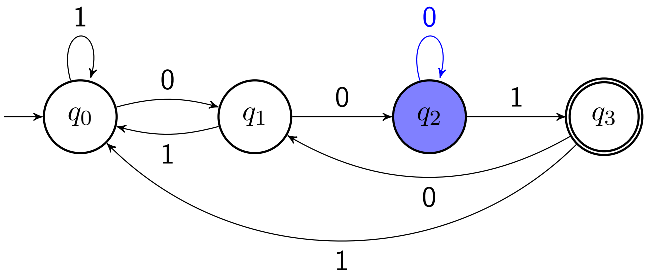
Introduction



1 0 1 0 0 1 0 0 0 1

↑

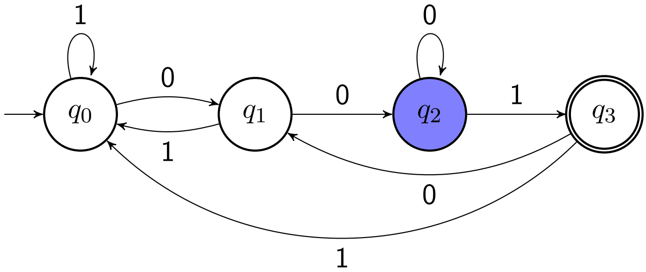
Introduction



1 0 1 0 0 1 0 0 0 1

↑

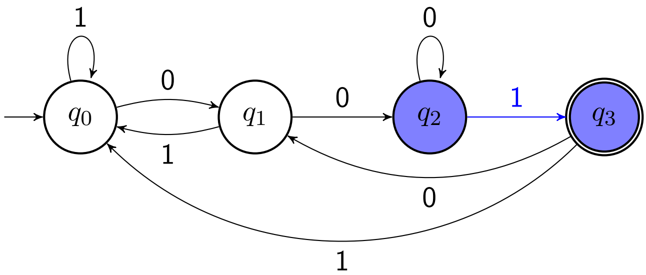
Introduction



1 0 1 0 0 1 0 0 0 1

↑

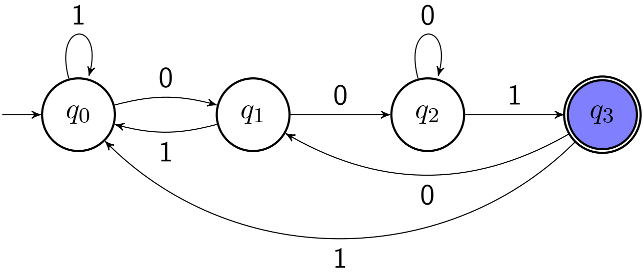
Introduction



1 0 1 0 0 1 0 0 0 1

↑

Introduction



1 0 1 0 0 1 0 0 0 1

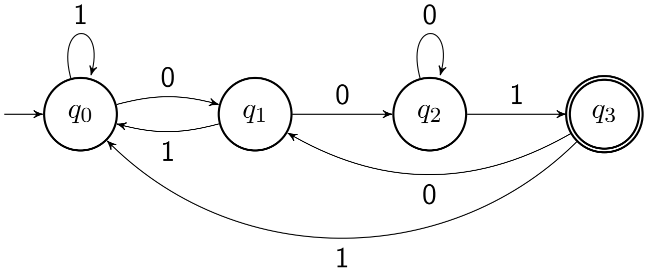
↑

Deterministic finite state automata

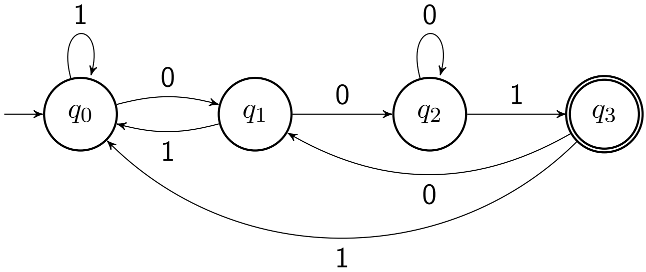
A *deterministic finite state automaton* is a 5-tuple $A = \langle Q, \Sigma, \delta, q_0, F \rangle$, where

- Q is an alphabet, the elements of which are called *states*;
- Σ is an alphabet, the elements of which are called *input symbols*;
- $\delta \in Q^{Q \times \Sigma}$ is called the *transition function*;
- $q_0 \in Q$ is called the *initial state*;
- $F \subset Q$ is a set of *final states*.

Deterministic finite state automata



Deterministic finite state automata



$$Q = \{q_0, q_1, q_2, q_3\}$$
$$\Sigma = \{0, 1\}$$
$$F = \{q_3\}$$

$$\delta(q_0, 0) = q_1 \quad \delta(q_2, 0) = q_2$$
$$\delta(q_0, 1) = q_0 \quad \delta(q_2, 1) = q_3$$
$$\delta(q_1, 0) = q_2 \quad \delta(q_3, 0) = q_1$$
$$\delta(q_1, 1) = q_0 \quad \delta(q_3, 1) = q_0$$

Deterministic finite state automata

Extended transition function:

$$\hat{\delta} \in Q^{Q \times \Sigma^*}$$

$$\begin{aligned} \hat{\delta}(q, \epsilon) &= q \\ \hat{\delta}(q, a\alpha) &= \hat{\delta}(\delta(q, a), \alpha) \end{aligned}$$

Deterministic finite state automata

Extended transition function:

$$\hat{\delta} \in Q^{Q \times \Sigma^*}$$

$$\begin{array}{rcl} \hat{\delta}(q, \epsilon) & = & q \\ \hat{\delta}(q, a\alpha) & = & \hat{\delta}(\delta(q, a), \alpha) \end{array}$$

Language accepted by a DFSA:

$$L(A) = \{\alpha \in \Sigma^* : \hat{\delta}(q_0, \alpha) \in F\}$$

Exercice

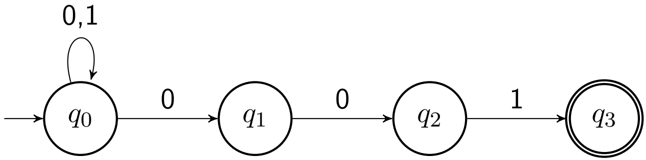
Design a DFSA that recognizes the natural numbers divisible by 3.

Non-deterministic finite state automata

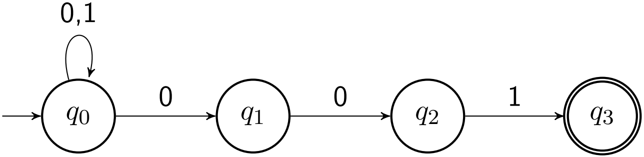
A *non-deterministic finite state automaton* is a 5-tuple $A = \langle Q, \Sigma, \delta, q_0, F \rangle$, where

- Q is an alphabet, the elements of which are called *states*;
- Σ is an alphabet, the elements of which are called *input symbols*;
- $\delta \in (2^Q)^{Q \times \Sigma}$ is a *transition function*;
- $q_0 \in Q$ is called the *initial state*;
- $F \subset Q$ is a set of *final states*.

Non-deterministic finite state automata



Non-deterministic finite state automata



$Q = \{q_0, q_1, q_2, q_3\}$
$\Sigma = \{0, 1\}$
$F = \{q_3\}$

$\delta(q_0, 0) = \{q_0, q_1\}$
$\delta(q_0, 1) = \{q_0\}$
$\delta(q_1, 0) = \{q_2\}$
$\delta(q_1, 1) = \varnothing$

$\delta(q_2, 0) = \varnothing$
$\delta(q_2, 1) = \{q_3\}$
$\delta(q_3, 0) = \varnothing$
$\delta(q_3, 1) = \varnothing$

Non-deterministic finite state automata

Extended transition function:

$$\hat{\delta} \in (2^Q)^{Q \times \Sigma^*}$$

$$\begin{array}{rcl} \hat{\delta}(q, \epsilon) & = & \{q\} \\ \hat{\delta}(q, a\alpha) & = & \bigcup_{p \in \delta(q,a)} \hat{\delta}(p, \alpha) \end{array}$$

Non-deterministic finite state automata

Extended transition function:

$$\hat{\delta} \in (2^Q)^{Q \times \Sigma^*}$$

$$\begin{array}{rcl} \hat{\delta}(q, \epsilon) & = & \{q\} \\ \hat{\delta}(q, a\alpha) & = & \bigcup_{p \in \delta(q,a)} \hat{\delta}(p, \alpha) \end{array}$$

Language accepted by an NFSA:

$$L(A) = \{\alpha \in \Sigma^* : \hat{\delta}(q_0, \alpha) \cap F \neq \varnothing\}$$

Non-deterministic finite state automata

Extended transition function:

$$\hat{\delta} \in (2^Q)^{Q \times \Sigma^*}$$

$$\begin{aligned} \hat{\delta}(q, \epsilon) &= \{q\} \\ \hat{\delta}(q, a\alpha) &= \bigcup_{p \in \delta(q,a)} \hat{\delta}(p, \alpha) \end{aligned}$$

Language accepted by an NFSA:

$$L(A) = \{\alpha \in \Sigma^* : \hat{\delta}(q_0, \alpha) \cap F \neq \varnothing\}$$

Remark: every DFSA $A = \langle Q, \Sigma, \delta, q_0, F \rangle$ may be seen as an NFSA $A' = \langle Q, \Sigma, \delta', q_0, F \rangle$ by defining $\delta'(q, a) = \{\delta(q, a)\}$.

Equivalence of NFSA and DFSA

Let $A_N = \langle Q_N, \Sigma_N, \delta_N, q_N, F_N \rangle$ be an NFSA. Define a DFSA $A_D = \langle Q_D, \Sigma_D, \delta_D, q_D, F_D \rangle$ as follows:

- $Q_D = 2^{Q_N}$
- $\Sigma_D = \Sigma_N$
- $\delta_D(S, a) = \bigcup_{p \in S} \delta_N(p, a)$
- $q_D = \{q_N\}$
- $F_D = \{S \subset Q : S \cap F_N \neq \varnothing\}$

Equivalence of NFSA and DFSA

Let $A_N = \langle Q_N, \Sigma_N, \delta_N, q_N, F_N \rangle$ be an NFSA. Define a DFSA $A_D = \langle Q_D, \Sigma_D, \delta_D, q_D, F_D \rangle$ as follows:

- $Q_D = 2^{Q_N}$
- $\Sigma_D = \Sigma_N$
- $\delta_D(S, a) = \bigcup_{p \in S} \delta_N(p, a)$
- $q_D = \{q_N\}$
- $F_D = \{S \subset Q : S \cap F_N \neq \varnothing\}$

Proposition $L(A_N) = L(A_D)$.

Equivalence of NFSA and DFSA

Let $A_N = \langle Q_N, \Sigma_N, \delta_N, q_N, F_N \rangle$ be an NFSA. Define a DFSA $A_D = \langle Q_D, \Sigma_D, \delta_D, q_D, F_D \rangle$ as follows:

- $Q_D = 2^{Q_N}$
- $\Sigma_D = \Sigma_N$
- $\delta_D(S, a) = \bigcup_{p \in S} \delta_N(p, a)$
- $q_D = \{q_N\}$
- $F_D = \{S \subset Q : S \cap F_N \neq \varnothing\}$

Proposition $L(A_N) = L(A_D)$.

Corollary A language is accepted by an NFSA if and only if it is accepted by a DFSA.

Equivalence of NFSA and DFSA

PROOF:

Equivalence of NFSA and DFSA

PROOF:

We prove by induction on the length of α that $\hat{\delta}_D(S, \alpha) = \bigcup_{q \in S} \hat{\delta}_N(q, \alpha)$

Equivalence of NFSA and DFSA

PROOF:

We prove by induction on the length of α that $\hat{\delta}_D(S, \alpha) = \bigcup_{q \in S} \hat{\delta}_N(q, \alpha)$

Basis:

Equivalence of NFSA and DFSA

PROOF:

We prove by induction on the length of α that $\hat{\delta}_D(S, \alpha) = \bigcup_{q \in S} \hat{\delta}_N(q, \alpha)$

Basis:

$$\hat{\delta}_D(S, \epsilon) = S$$

Equivalence of NFSA and DFSA

PROOF:

We prove by induction on the length of α that $\hat{\delta}_D(S, \alpha) = \bigcup_{q \in S} \hat{\delta}_N(q, \alpha)$

Basis:

$$\begin{aligned} \hat{\delta}_D(S, \epsilon) &= S \\ &= \bigcup_{q \in S} \{q\} \end{aligned}$$

Equivalence of NFSA and DFSA

PROOF:

We prove by induction on the length of α that $\hat{\delta}_D(S,\alpha) = \bigcup_{q\in S} \hat{\delta}_N(q,\alpha)$

Basis:

$$\begin{aligned} \hat{\delta}_D(S,\epsilon) &= S \\ &= \bigcup_{q\in S}\{q\} \\ &= \bigcup_{q\in S} \hat{\delta}_N(q,\epsilon) \end{aligned}$$

Equivalence of NFSA and DFSA

PROOF:

We prove by induction on the length of α that $\hat{\delta}_D(S, \alpha) = \bigcup_{q \in S} \hat{\delta}_N(q, \alpha)$

Basis:

$$\begin{aligned} \hat{\delta}_D(S, \epsilon) &= S \\ &= \bigcup_{q \in S} \{q\} \\ &= \bigcup_{q \in S} \hat{\delta}_N(q, \epsilon) \end{aligned}$$

Induction:

Equivalence of NFSA and DFSA

PROOF:

We prove by induction on the length of α that $\hat{\delta}_D(S, \alpha) = \bigcup_{q \in S} \hat{\delta}_N(q, \alpha)$

Basis:

$$\begin{aligned}\hat{\delta}_D(S, \epsilon) &= S \\ &= \bigcup_{q \in S} \{q\} \\ &= \bigcup_{q \in S} \hat{\delta}_N(q, \epsilon)\end{aligned}$$

Induction:

$$\hat{\delta}_D(S, a\alpha') = \hat{\delta}_D(\delta_D(S, a), \alpha')$$

Equivalence of NFSA and DFSA

PROOF:

We prove by induction on the length of α that $\hat{\delta}_D(S,\alpha) = \bigcup_{q\in S} \hat{\delta}_N(q,\alpha)$

Basis:

$$\begin{aligned}\hat{\delta}_D(S,\epsilon) &= S \\ &= \textstyle\bigcup_{q\in S}\{q\} \\ &= \textstyle\bigcup_{q\in S} \hat{\delta}_N(q,\epsilon)\end{aligned}$$

Induction:

$$\begin{aligned}\hat{\delta}_D(S,a\alpha') &= \hat{\delta}_D(\delta_D(S,a),\alpha') \\ &= \hat{\delta}_D(\textstyle\bigcup_{q\in S} \delta_N(q,a),\alpha') && \text{by definition of } \delta_D\end{aligned}$$

Equivalence of NFSA and DFSA

PROOF:

We prove by induction on the length of α that $\hat{\delta}_D(S, \alpha) = \bigcup_{q \in S} \hat{\delta}_N(q, \alpha)$

Basis:

$$\begin{aligned} \hat{\delta}_D(S, \epsilon) &= S \\ &= \bigcup_{q \in S} \{q\} \\ &= \bigcup_{q \in S} \hat{\delta}_N(q, \epsilon) \end{aligned}$$

Induction:

$$\begin{aligned} \hat{\delta}_D(S, a\alpha') &= \hat{\delta}_D(\delta_D(S, a), \alpha') \\ &= \hat{\delta}_D(\bigcup_{q \in S} \delta_N(q, a), \alpha') && \text{by definition of } \delta_D \\ &= \bigcup_{p \in \bigcup_{q \in S} \delta_N(q,a)} \hat{\delta}_N(p, \alpha') && \text{by induction hypothesis} \end{aligned}$$

Equivalence of NFSA and DFSA

PROOF:

We prove by induction on the length of α that $\hat{\delta}_D(S,\alpha) = \bigcup_{q\in S} \hat{\delta}_N(q,\alpha)$

Basis:

$$\begin{aligned}\hat{\delta}_D(S,\epsilon) &= S \\ &= \textstyle\bigcup_{q\in S}\{q\} \\ &= \textstyle\bigcup_{q\in S} \hat{\delta}_N(q,\epsilon)\end{aligned}$$

Induction:

$$\begin{aligned}\hat{\delta}_D(S,a\alpha') &= \hat{\delta}_D(\delta_D(S,a),\alpha') \\ &= \hat{\delta}_D(\textstyle\bigcup_{q\in S} \delta_N(q,a),\alpha') && \text{by definition of } \delta_D \\ &= \textstyle\bigcup_{p\in \bigcup_{q\in S} \delta_N(q,a)} \hat{\delta}_N(p,\alpha') && \text{by induction hypothesis} \\ &= \textstyle\bigcup_{q\in S}(\bigcup_{p\in \delta_N(q,a)} \hat{\delta}_N(p,\alpha'))\end{aligned}$$

Equivalence of NFSA and DFSA

PROOF:

We prove by induction on the length of α that $\hat{\delta}_D(S,\alpha) = \bigcup_{q\in S} \hat{\delta}_N(q,\alpha)$

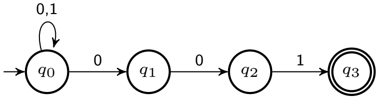
Basis:

$$\begin{aligned}\hat{\delta}_D(S,\epsilon) &= S \\ &= \bigcup_{q\in S}\{q\} \\ &= \bigcup_{q\in S} \hat{\delta}_N(q,\epsilon)\end{aligned}$$

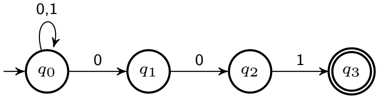
Induction:

$$\begin{aligned}\hat{\delta}_D(S,a\alpha') &= \hat{\delta}_D(\delta_D(S,a),\alpha') \\ &= \hat{\delta}_D(\bigcup_{q\in S} \delta_N(q,a),\alpha') && \text{by definition of } \delta_D \\ &= \bigcup_{p\in\bigcup_{q\in S}\delta_N(q,a)} \hat{\delta}_N(p,\alpha') && \text{by induction hypothesis} \\ &= \bigcup_{q\in S}(\bigcup_{p\in\delta_N(q,a)} \hat{\delta}_N(p,\alpha')) \\ &= \bigcup_{q\in S}(\hat{\delta}(q,a\alpha'))\end{aligned}$$

Equivalence of NFSA and DFSA



Equivalence of NFSA and DFSA

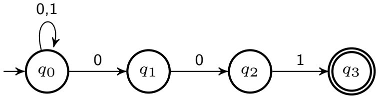


$$\delta_d(\varnothing,0)=\varnothing$$
$$\delta_d(\varnothing,1)=\varnothing$$
$$\delta_d(\{q_0\},0)=\{q_0,q_1\}$$
$$\delta_d(\{q_0\},1)=\{q_0\}$$
$$\delta_d(\{q_1\},0)=\{q_2\}$$
$$\delta_d(\{q_1\},1)=\varnothing$$
$$\delta_d(\{q_2\},0)=\varnothing$$
$$\delta_d(\{q_2\},1)=\{q_3\}$$
$$\delta_d(\{q_3\},0)=\varnothing$$
$$\delta_d(\{q_3\},1)=\varnothing$$
$$\delta_d(\{q_0,q_1\},0)=\{q_0,q_1,q_2\}$$
$$\delta_d(\{q_0,q_1\},1)=\{q_0\}$$

$$\delta_d(\{q_0,q_2\},0)=\{q_0,q_1\}$$
$$\delta_d(\{q_0,q_2\},1)=\{q_0,q_3\}$$
$$\delta_d(\{q_0,q_3\},0)=\{q_0,q_1\}$$
$$\delta_d(\{q_0,q_3\},1)=\{q_0\}$$
$$\delta_d(\{q_1,q_2\},0)=\{q_2\}$$
$$\delta_d(\{q_1,q_2\},1)=\{q_3\}$$
$$\delta_d(\{q_1,q_3\},0)=\{q_2\}$$
$$\delta_d(\{q_1,q_3\},1)=\varnothing$$
$$\delta_d(\{q_2,q_3\},0)=\varnothing$$
$$\delta_d(\{q_2,q_3\},1)=\{q_3\}$$
$$\delta_d(\{q_0,q_1,q_2\},0)=\{q_0,q_1,q_2\}$$
$$\delta_d(\{q_0,q_1,q_2\},1)=\{q_0,q_3\}$$

$$\delta_d(\{q_0,q_1,q_3\},0)=\{q_0,q_1,q_2\}$$
$$\delta_d(\{q_0,q_1,q_3\},1)=\{q_0\}$$
$$\delta_d(\{q_0,q_2,q_3\},0)=\{q_0,q_1\}$$
$$\delta_d(\{q_0,q_2,q_3\},1)=\{q_0,q_3\}$$
$$\delta_d(\{q_1,q_2,q_3\},0)=\{q_2\}$$
$$\delta_d(\{q_1,q_2,q_3\},1)=\{q_3\}$$
$$\delta_d(\{q_0,q_1,q_2,q_3\},0)=\{q_0,q_1,q_2\}$$
$$\delta_d(\{q_0,q_1,q_2,q_3\},1)=\{q_0,q_3\}$$

Equivalence of NFSA and DFSA

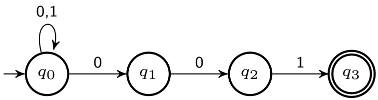


$\delta_d(\emptyset, 0) = \emptyset$
$\delta_d(\emptyset, 1) = \emptyset$
$\delta_d(\{q_0\}, 0) = \{q_0, q_1\}$
$\delta_d(\{q_0\}, 1) = \{q_0\}$
$\delta_d(\{q_1\}, 0) = \{q_2\}$
$\delta_d(\{q_1\}, 1) = \emptyset$
$\delta_d(\{q_2\}, 0) = \emptyset$
$\delta_d(\{q_2\}, 1) = \{q_3\}$
$\delta_d(\{q_3\}, 0) = \emptyset$
$\delta_d(\{q_3\}, 1) = \emptyset$
$\delta_d(\{q_0, q_1\}, 0) = \{q_0, q_1, q_2\}$
$\delta_d(\{q_0, q_1\}, 1) = \{q_0\}$

$\delta_d(\{q_0, q_2\}, 0) = \{q_0, q_1\}$
$\delta_d(\{q_0, q_2\}, 1) = \{q_0, q_3\}$
$\delta_d(\{q_0, q_3\}, 0) = \{q_0, q_1\}$
$\delta_d(\{q_0, q_3\}, 1) = \{q_0\}$
$\delta_d(\{q_1, q_2\}, 0) = \{q_2\}$
$\delta_d(\{q_1, q_2\}, 1) = \{q_3\}$
$\delta_d(\{q_1, q_3\}, 0) = \{q_2\}$
$\delta_d(\{q_1, q_3\}, 1) = \emptyset$
$\delta_d(\{q_2, q_3\}, 0) = \emptyset$
$\delta_d(\{q_2, q_3\}, 1) = \{q_3\}$
$\delta_d(\{q_0, q_1, q_2\}, 0) = \{q_0, q_1, q_2\}$
$\delta_d(\{q_0, q_1, q_2\}, 1) = \{q_0, q_3\}$

$\delta_d(\{q_0, q_1, q_3\}, 0) = \{q_0, q_1, q_2\}$
$\delta_d(\{q_0, q_1, q_3\}, 1) = \{q_0\}$
$\delta_d(\{q_0, q_2, q_3\}, 0) = \{q_0, q_1\}$
$\delta_d(\{q_0, q_2, q_3\}, 1) = \{q_0, q_3\}$
$\delta_d(\{q_1, q_2, q_3\}, 0) = \{q_2\}$
$\delta_d(\{q_1, q_2, q_3\}, 1) = \{q_3\}$
$\delta_d(\{q_0, q_1, q_2, q_3\}, 0) = \{q_0, q_1, q_2\}$
$\delta_d(\{q_0, q_1, q_2, q_3\}, 1) = \{q_0, q_3\}$

Equivalence of NFSA and DFSA

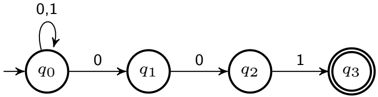


$\delta_d(\varnothing, 0) = \varnothing$
$\delta_d(\varnothing, 1) = \varnothing$
$\delta_d(\{q_0\}, 0) = \{q_0, q_1\}$
$\delta_d(\{q_0\}, 1) = \{q_0\}$
$\delta_d(\{q_1\}, 0) = \{q_2\}$
$\delta_d(\{q_1\}, 1) = \varnothing$
$\delta_d(\{q_2\}, 0) = \varnothing$
$\delta_d(\{q_2\}, 1) = \{q_3\}$
$\delta_d(\{q_3\}, 0) = \varnothing$
$\delta_d(\{q_3\}, 1) = \varnothing$
$\delta_d(\{q_0, q_1\}, 0) = \{q_0, q_1, q_2\}$
$\delta_d(\{q_0, q_1\}, 1) = \{q_0\}$

$\delta_d(\{q_0, q_2\}, 0) = \{q_0, q_1\}$
$\delta_d(\{q_0, q_2\}, 1) = \{q_0, q_3\}$
$\delta_d(\{q_0, q_3\}, 0) = \{q_0, q_1\}$
$\delta_d(\{q_0, q_3\}, 1) = \{q_0\}$
$\delta_d(\{q_1, q_2\}, 0) = \{q_2\}$
$\delta_d(\{q_1, q_2\}, 1) = \{q_3\}$
$\delta_d(\{q_1, q_3\}, 0) = \{q_2\}$
$\delta_d(\{q_1, q_3\}, 1) = \varnothing$
$\delta_d(\{q_2, q_3\}, 0) = \varnothing$
$\delta_d(\{q_2, q_3\}, 1) = \{q_3\}$
$\delta_d(\{q_0, q_1, q_2\}, 0) = \{q_0, q_1, q_2\}$
$\delta_d(\{q_0, q_1, q_2\}, 1) = \{q_0, q_3\}$

$\delta_d(\{q_0, q_1, q_3\}, 0) = \{q_0, q_1, q_2\}$
$\delta_d(\{q_0, q_1, q_3\}, 1) = \{q_0\}$
$\delta_d(\{q_0, q_2, q_3\}, 0) = \{q_0, q_1\}$
$\delta_d(\{q_0, q_2, q_3\}, 1) = \{q_0, q_3\}$
$\delta_d(\{q_1, q_2, q_3\}, 0) = \{q_2\}$
$\delta_d(\{q_1, q_2, q_3\}, 1) = \{q_3\}$
$\delta_d(\{q_0, q_1, q_2, q_3\}, 0) = \{q_0, q_1, q_2\}$
$\delta_d(\{q_0, q_1, q_2, q_3\}, 1) = \{q_0, q_3\}$

Equivalence of NFSA and DFSA

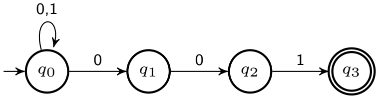


$\delta_d(\varnothing, 0) = \varnothing$
$\delta_d(\varnothing, 1) = \varnothing$
$\delta_d(\{q_0\}, 0) = \{q_0, q_1\}$
$\delta_d(\{q_0\}, 1) = \{q_0\}$
$\delta_d(\{q_1\}, 0) = \{q_2\}$
$\delta_d(\{q_1\}, 1) = \varnothing$
$\delta_d(\{q_2\}, 0) = \varnothing$
$\delta_d(\{q_2\}, 1) = \{q_3\}$
$\delta_d(\{q_3\}, 0) = \varnothing$
$\delta_d(\{q_3\}, 1) = \varnothing$
$\delta_d(\{q_0, q_1\}, 0) = \{q_0, q_1, q_2\}$
$\delta_d(\{q_0, q_1\}, 1) = \{q_0\}$

$\delta_d(\{q_0, q_2\}, 0) = \{q_0, q_1\}$
$\delta_d(\{q_0, q_2\}, 1) = \{q_0, q_3\}$
$\delta_d(\{q_0, q_3\}, 0) = \{q_0, q_1\}$
$\delta_d(\{q_0, q_3\}, 1) = \{q_0\}$
$\delta_d(\{q_1, q_2\}, 0) = \{q_2\}$
$\delta_d(\{q_1, q_2\}, 1) = \{q_3\}$
$\delta_d(\{q_1, q_3\}, 0) = \{q_2\}$
$\delta_d(\{q_1, q_3\}, 1) = \varnothing$
$\delta_d(\{q_2, q_3\}, 0) = \varnothing$
$\delta_d(\{q_2, q_3\}, 1) = \{q_3\}$
$\delta_d(\{q_0, q_1, q_2\}, 0) = \{q_0, q_1, q_2\}$
$\delta_d(\{q_0, q_1, q_2\}, 1) = \{q_0, q_3\}$

$\delta_d(\{q_0, q_1, q_3\}, 0) = \{q_0, q_1, q_2\}$
$\delta_d(\{q_0, q_1, q_3\}, 1) = \{q_0\}$
$\delta_d(\{q_0, q_2, q_3\}, 0) = \{q_0, q_1\}$
$\delta_d(\{q_0, q_2, q_3\}, 1) = \{q_0, q_3\}$
$\delta_d(\{q_1, q_2, q_3\}, 0) = \{q_2\}$
$\delta_d(\{q_1, q_2, q_3\}, 1) = \{q_3\}$
$\delta_d(\{q_0, q_1, q_2, q_3\}, 0) = \{q_0, q_1, q_2\}$
$\delta_d(\{q_0, q_1, q_2, q_3\}, 1) = \{q_0, q_3\}$

Equivalence of NFSA and DFSA

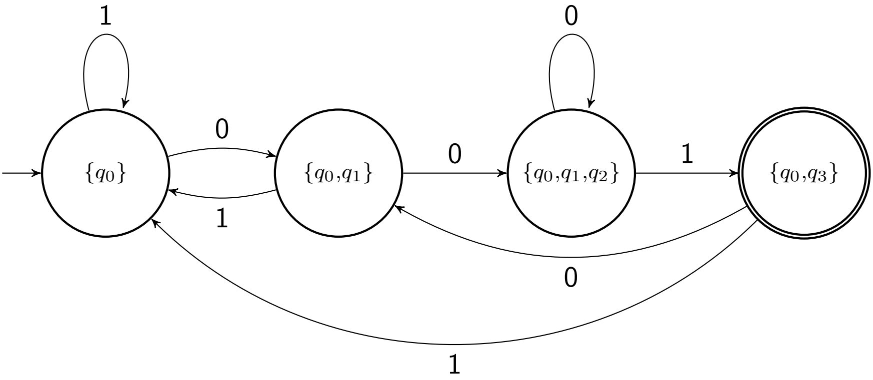


$\delta_d(\varnothing, 0) = \varnothing$
$\delta_d(\varnothing, 1) = \varnothing$
$\delta_d(\{q_0\}, 0) = \{q_0, q_1\}$
$\delta_d(\{q_0\}, 1) = \{q_0\}$
$\delta_d(\{q_1\}, 0) = \{q_2\}$
$\delta_d(\{q_1\}, 1) = \varnothing$
$\delta_d(\{q_2\}, 0) = \varnothing$
$\delta_d(\{q_2\}, 1) = \{q_3\}$
$\delta_d(\{q_3\}, 0) = \varnothing$
$\delta_d(\{q_3\}, 1) = \varnothing$
$\delta_d(\{q_0, q_1\}, 0) = \{q_0, q_1, q_2\}$
$\delta_d(\{q_0, q_1\}, 1) = \{q_0\}$

$\delta_d(\{q_0, q_2\}, 0) = \{q_0, q_1\}$
$\delta_d(\{q_0, q_2\}, 1) = \{q_0, q_3\}$
$\delta_d(\{q_0, q_3\}, 0) = \{q_0, q_1\}$
$\delta_d(\{q_0, q_3\}, 1) = \{q_0\}$
$\delta_d(\{q_1, q_2\}, 0) = \{q_2\}$
$\delta_d(\{q_1, q_2\}, 1) = \{q_3\}$
$\delta_d(\{q_1, q_3\}, 0) = \{q_2\}$
$\delta_d(\{q_1, q_3\}, 1) = \varnothing$
$\delta_d(\{q_2, q_3\}, 0) = \varnothing$
$\delta_d(\{q_2, q_3\}, 1) = \{q_3\}$
$\delta_d(\{q_0, q_1, q_2\}, 0) = \{q_0, q_1, q_2\}$
$\delta_d(\{q_0, q_1, q_2\}, 1) = \{q_0, q_3\}$

$\delta_d(\{q_0, q_1, q_3\}, 0) = \{q_0, q_1, q_2\}$
$\delta_d(\{q_0, q_1, q_3\}, 1) = \{q_0\}$
$\delta_d(\{q_0, q_2, q_3\}, 0) = \{q_0, q_1\}$
$\delta_d(\{q_0, q_2, q_3\}, 1) = \{q_0, q_3\}$
$\delta_d(\{q_1, q_2, q_3\}, 0) = \{q_2\}$
$\delta_d(\{q_1, q_2, q_3\}, 1) = \{q_3\}$
$\delta_d(\{q_0, q_1, q_2, q_3\}, 0) = \{q_0, q_1, q_2\}$
$\delta_d(\{q_0, q_1, q_2, q_3\}, 1) = \{q_0, q_3\}$

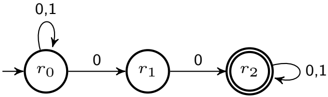
Equivalence of NFSA and DFSA



Finite state automata with ϵ-transitions

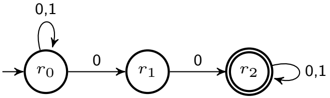
Finite state automata with ϵ-transitions

An automaton that recognizes binary words with at least two consecutive 0's.

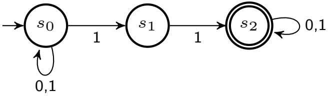


Finite state automata with ϵ-transitions

An automaton that recognizes binary words with at least two consecutive 0's.

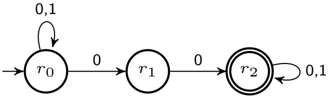


An automaton that recognizes binary words with at least two consecutive 1's.

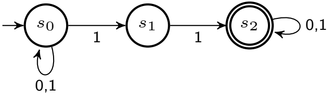


Finite state automata with ϵ-transitions

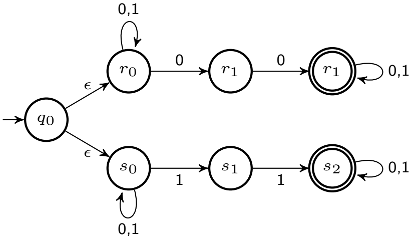
An automaton that recognizes binary words with at least two consecutive 0's.



An automaton that recognizes binary words with at least two consecutive 1's.



An automaton that recognizes binary words with at least two consecutive identical digits.



Finite state automata with ϵ-transitions

A *finite state automaton with ϵ-transitions* is a 5-tuple $A = \langle Q, \Sigma, \delta, q_0, F \rangle$, where:

- Q is an alphabet, the elements of which are called *states*;
- Σ is an alphabet, the elements of which are called *input symbols*;
- $\delta \in (2^Q)^{Q \times (\Sigma \cup \{\epsilon\})}$ is a *transition function*;
- $q_0 \in Q$ is called the *initial state*;
- $F \subset Q$ is a set of *final states*.

Finite state automata with ϵ-transitions

ϵ-closure of a state q:

- $q \in Cl_\epsilon(q)$;
- if $p \in Cl_\epsilon(q)$ and $s \in \delta(p, \epsilon)$ then $s \in Cl_\epsilon(q)$.

Finite state automata with ϵ-transitions

ϵ-closure of a state q:

- $q \in Cl_\epsilon(q)$;
- if $p \in Cl_\epsilon(q)$ and $s \in \delta(p, \epsilon)$ then $s \in Cl_\epsilon(q)$.

Extended transition function:

$$\hat{\delta} \in (2^Q)^{Q \times \Sigma^*}$$

$$\begin{array}{rcl} \hat{\delta}(q, \epsilon) & = & Cl_\epsilon(q) \\ \hat{\delta}(q, a\alpha) & = & \bigcup_{p \in Cl_\epsilon(q)} \bigcup_{s \in \delta(p,a)} \hat{\delta}(s, \alpha) \end{array}$$

Finite state automata with ϵ-transitions

ϵ-closure of a state q:

- $q \in Cl_\epsilon(q)$;
- if $p \in Cl_\epsilon(q)$ and $s \in \delta(p, \epsilon)$ then $s \in Cl_\epsilon(q)$.

Extended transition function:

$$\hat{\delta} \in (2^Q)^{Q \times \Sigma^*}$$

$$\begin{array}{rcl} \hat{\delta}(q, \epsilon) & = & Cl_\epsilon(q) \\ \hat{\delta}(q, a\alpha) & = & \bigcup_{p \in Cl_\epsilon(q)} \bigcup_{s \in \delta(p,a)} \hat{\delta}(s, \alpha) \end{array}$$

Language accepted by a ϵ-FSA:

$$L(A) = \{\alpha \in \Sigma^* : \hat{\delta}(q_0, \alpha) \cap F \neq \varnothing\}$$

Elimination of ϵ-transitions

Let $A_E = \langle Q_E, \Sigma_E, \delta_E, q_E, F_E \rangle$ be an ϵ-FSA. Define an NFSA $A_N = \langle Q_N, \Sigma_N, \delta_N, q_N, F_N \rangle$ as follows:

- $Q_N = Q_E$
- $\Sigma_N = \Sigma_E$
- $\delta_N(q, a) = \bigcup_{p \in Cl_\epsilon(q)} \delta_E(p, a)$
- $q_N = q_E$
- $F_N = \{q \in Q_E : Cl_\epsilon(q) \cap F_E \neq \varnothing\}$

Elimination of ϵ-transitions

Let $A_E = \langle Q_E, \Sigma_E, \delta_E, q_E, F_E \rangle$ be an ϵ-FSA. Define an NFSA $A_N = \langle Q_N, \Sigma_N, \delta_N, q_N, F_N \rangle$ as follows:

- $Q_N = Q_E$
- $\Sigma_N = \Sigma_E$
- $\delta_N(q, a) = \bigcup_{p \in Cl_\epsilon(q)} \delta_E(p, a)$
- $q_N = q_E$
- $F_N = \{q \in Q_E : Cl_\epsilon(q) \cap F_E \neq \varnothing\}$

Proposition $L(A_E) = L(A_N)$.

Elimination of ϵ-transitions

Let $A_E = \langle Q_E, \Sigma_E, \delta_E, q_E, F_E \rangle$ be an ϵ-FSA. Define an NFSA $A_N = \langle Q_N, \Sigma_N, \delta_N, q_N, F_N \rangle$ as follows:

- $Q_N = Q_E$
- $\Sigma_N = \Sigma_E$
- $\delta_N(q, a) = \bigcup_{p \in Cl_\epsilon(q)} \delta_E(p, a)$
- $q_N = q_E$
- $F_N = \{q \in Q_E : Cl_\epsilon(q) \cap F_E \neq \varnothing\}$

Proposition $L(A_E) = L(A_N)$.

Corollary A language is accepted by an ϵ-FSA if and only if it is accepted by an NFSA.

Elimination of ϵ-transitions

PROOF:

Elimination of ϵ-transitions

PROOF:

We prove by induction on the length of α that $\hat{\delta}_N(q, \alpha) \cap F_N \neq \varnothing$ if and only if $\hat{\delta}_E(q, \alpha) \cap F_E \neq \varnothing$.

Elimination of ϵ-transitions

PROOF:

We prove by induction on the length of α that $\hat{\delta}_N(q, \alpha) \cap F_N \neq \varnothing$ if and only if $\hat{\delta}_E(q, \alpha) \cap F_E \neq \varnothing$.

Basis:

Elimination of ϵ-transitions

PROOF:

We prove by induction on the length of α that $\hat{\delta}_N(q, \alpha) \cap F_N \neq \varnothing$ if and only if $\hat{\delta}_E(q, \alpha) \cap F_E \neq \varnothing$.

Basis:

$\hat{\delta}_N(q, \epsilon) \cap F_N \neq \varnothing$ iff $\{q\} \cap F_N \neq \varnothing$

Elimination of ϵ-transitions

PROOF:

We prove by induction on the length of α that $\hat{\delta}_N(q, \alpha) \cap F_N \neq \varnothing$ if and only if $\hat{\delta}_E(q, \alpha) \cap F_E \neq \varnothing$.

Basis:

$$\begin{aligned} \hat{\delta}_N(q, \epsilon) \cap F_N \neq \varnothing & \text{ iff } \{q\} \cap F_N \neq \varnothing \\ & \text{ iff } q \in F_N \end{aligned}$$

Elimination of ϵ-transitions

PROOF:

We prove by induction on the length of α that $\hat{\delta}_N(q,\alpha) \cap F_N \neq \varnothing$ if and only if $\hat{\delta}_E(q,\alpha) \cap F_E \neq \varnothing$.

Basis:

$$\begin{aligned} \hat{\delta}_N(q,\epsilon) \cap F_N \neq \varnothing \ & \text{iff } \{q\} \cap F_N \neq \varnothing \\ & \text{iff } q \in F_N \\ & \text{iff } Cl_\epsilon(q) \cap F_E \neq \varnothing \end{aligned}$$

Elimination of ϵ-transitions

PROOF:

We prove by induction on the length of α that $\hat{\delta}_N(q, \alpha) \cap F_N \neq \varnothing$ if and only if $\hat{\delta}_E(q, \alpha) \cap F_E \neq \varnothing$.

Basis:

$$
\begin{aligned}
\hat{\delta}_N(q, \epsilon) \cap F_N \neq \varnothing \;\; & \text{iff } \{q\} \cap F_N \neq \varnothing \\
& \text{iff } q \in F_N \\
& \text{iff } Cl_\epsilon(q) \cap F_E \neq \varnothing \\
& \text{iff } \hat{\delta}_E(q, \epsilon) \cap F_E \neq \varnothing
\end{aligned}
$$

Elimination of ϵ-transitions

Induction:

Elimination of ϵ-transitions

Induction:

$\hat{\delta}_N(q, a\alpha') \cap F_N \neq \varnothing$ iff $(\bigcup_{s \in \delta_N(q,a)} \hat{\delta}_N(s, \alpha')) \cap F_N \neq \varnothing$

Elimination of ϵ-transitions

Induction:

$\hat{\delta}_N(q, a\alpha') \cap F_N \neq \varnothing$ iff $(\bigcup_{s\in\delta_N(q,a)} \hat{\delta}_N(s, \alpha')) \cap F_N \neq \varnothing$

iff $(\bigcup_{s\in\bigcup_{p\in Cl_\epsilon(q)} \delta_E(p,a)} \hat{\delta}_N(s, \alpha')) \cap F_N \neq \varnothing$

by definition of δ_N

Elimination of ϵ-transitions

Induction:

$\hat{\delta}_N(q, a\alpha') \cap F_N \neq \varnothing$ iff $(\bigcup_{s \in \delta_N(q,a)} \hat{\delta}_N(s, \alpha')) \cap F_N \neq \varnothing$

iff $(\bigcup_{s \in \bigcup_{p \in Cl_\epsilon(q)} \delta_E(p,a)} \hat{\delta}_N(s, \alpha')) \cap F_N \neq \varnothing$

by definition of δ_N

iff $(\bigcup_{p \in Cl_\epsilon(q)} \bigcup_{s \in \delta_E(p,a)} \hat{\delta}_N(s, \alpha')) \cap F_N \neq \varnothing$

Elimination of ϵ-transitions

Induction:

$\hat{\delta}_N(q, a\alpha') \cap F_N \neq \varnothing$ iff $(\bigcup_{s \in \delta_N(q,a)} \hat{\delta}_N(s, \alpha')) \cap F_N \neq \varnothing$

iff $(\bigcup_{s \in \bigcup_{p \in Cl_\epsilon(q)} \delta_E(p,a)} \hat{\delta}_N(s, \alpha')) \cap F_N \neq \varnothing$

by definition of δ_N

iff $(\bigcup_{p \in Cl_\epsilon(q)} \bigcup_{s \in \delta_E(p,a)} \hat{\delta}_N(s, \alpha')) \cap F_N \neq \varnothing$

iff $\bigcup_{p \in Cl_\epsilon(q)} \bigcup_{s \in \delta_E(p,a)} (\hat{\delta}_N(s, \alpha') \cap F_N) \neq \varnothing$

Elimination of ϵ-transitions

Induction:

$\hat{\delta}_N(q, a\alpha') \cap F_N \neq \varnothing$ iff $(\bigcup_{s \in \delta_N(q,a)} \hat{\delta}_N(s, \alpha')) \cap F_N \neq \varnothing$

iff $(\bigcup_{s \in \bigcup_{p \in Cl_\epsilon(q)} \delta_E(p,a)} \hat{\delta}_N(s, \alpha')) \cap F_N \neq \varnothing$

by definition of δ_N

iff $(\bigcup_{p \in Cl_\epsilon(q)} \bigcup_{s \in \delta_E(p,a)} \hat{\delta}_N(s, \alpha')) \cap F_N \neq \varnothing$

iff $\bigcup_{p \in Cl_\epsilon(q)} \bigcup_{s \in \delta_E(p,a)} (\hat{\delta}_N(s, \alpha') \cap F_N) \neq \varnothing$

iff $\bigcup_{p \in Cl_\epsilon(q)} \bigcup_{s \in \delta_E(p,a)} (\hat{\delta}_E(s, \alpha') \cap F_E) \neq \varnothing$

by induction hypothesis

Elimination of ϵ-transitions

Induction:

$$\hat{\delta}_N(q, a\alpha') \cap F_N \neq \varnothing \text{ iff } (\bigcup_{s\in\delta_N(q,a)} \hat{\delta}_N(s,\alpha')) \cap F_N \neq \varnothing$$

$$\text{iff } (\bigcup_{s\in\bigcup_{p\in Cl_\epsilon(q)} \delta_E(p,a)} \hat{\delta}_N(s,\alpha')) \cap F_N \neq \varnothing$$

by definition of δ_N

$$\text{iff } (\bigcup_{p\in Cl_\epsilon(q)} \bigcup_{s\in\delta_E(p,a)} \hat{\delta}_N(s,\alpha')) \cap F_N \neq \varnothing$$

$$\text{iff } \bigcup_{p\in Cl_\epsilon(q)} \bigcup_{s\in\delta_E(p,a)} (\hat{\delta}_N(s,\alpha') \cap F_N) \neq \varnothing$$

$$\text{iff } \bigcup_{p\in Cl_\epsilon(q)} \bigcup_{s\in\delta_E(p,a)} (\hat{\delta}_E(s,\alpha') \cap F_E) \neq \varnothing$$

by induction hypothesis

$$\text{iff } (\bigcup_{p\in Cl_\epsilon(q)} \bigcup_{s\in\delta_E(p,a)} \hat{\delta}_E(s,\alpha')) \cap F_E \neq \varnothing$$

Elimination of ϵ-transitions

Induction:

$$\hat{\delta}_N(q, a\alpha') \cap F_N \neq \varnothing \text{ iff } (\bigcup_{s\in\delta_N(q,a)} \hat{\delta}_N(s,\alpha')) \cap F_N \neq \varnothing$$

$$\text{iff } (\bigcup_{s\in\bigcup_{p\in Cl_\epsilon(q)} \delta_E(p,a)} \hat{\delta}_N(s,\alpha')) \cap F_N \neq \varnothing$$

by definition of δ_N

$$\text{iff } (\bigcup_{p\in Cl_\epsilon(q)} \bigcup_{s\in\delta_E(p,a)} \hat{\delta}_N(s,\alpha')) \cap F_N \neq \varnothing$$

$$\text{iff } \bigcup_{p\in Cl_\epsilon(q)} \bigcup_{s\in\delta_E(p,a)} (\hat{\delta}_N(s,\alpha') \cap F_N) \neq \varnothing$$

$$\text{iff } \bigcup_{p\in Cl_\epsilon(q)} \bigcup_{s\in\delta_E(p,a)} (\hat{\delta}_E(s,\alpha') \cap F_E) \neq \varnothing$$

by induction hypothesis

$$\text{iff } (\bigcup_{p\in Cl_\epsilon(q)} \bigcup_{s\in\delta_E(p,a)} \hat{\delta}_E(s,\alpha')) \cap F_E \neq \varnothing$$

$$\text{iff } \hat{\delta}_E(q, a\alpha') \cap F_E \neq \varnothing$$