形式言語理論

Formal Language Theory

1. 有限オートマトン

九州大学
KYUSHU UNIVERSITY

本日の内容

- 準備
- 決定性有限オートマトン(DFA)
- 非決定性有限オートマトン(NFA)
- 拡張されたNFA

準備

アルファベット，文字列

- 文字(記号)の空でない有限集合をアルファベットとよぶ.
  - アルファベットは Σ で表すことが多い.
- アルファベット Σ の文字を有限個並べたものを(Σ上の)文字列という．特に，文字を 0 個並べたものを空文字列とよび，ε で表す.
- Σ上の文字列全体の集合を Σ^* で表し，空文字列以外の文字列全体の集合を Σ^+ で表す.
  - すなわち，$\Sigma^+ = \Sigma^* - \{\varepsilon\}$である.

文字列の連接

- 文字列 x, y を

  $x = a_1 \ldots a_n \qquad (n \geq 0; a_1, \ldots, a_n \in \Sigma)$

  $y = b_1 \ldots b_m \qquad (m \geq 0; b_1, \ldots, b_m \in \Sigma)$

  とするとき, x と y の**連接** $x \cdot y$ を次で定義する.

  $x \cdot y = a_1 \ldots a_n b_1 \ldots b_m$

- $x \cdot y$ を単に xy と書くことがある.
- 任意の $x \in \Sigma^*$ に対して $x \cdot \varepsilon = \varepsilon \cdot x = x$ が成り立つ.

文字列の長さ

- 文字列 $w = a_1 \ldots a_n$ $(n \geq 0; a_1, \ldots, a_n \in \Sigma)$に対し, 整数 n を w の長さといい, $|w|$で表す.

言語，積

- Σ^* の部分集合を Σ 上の言語という.
- 言語 L_1, L_2 に対して，L_1, L_2 の積(product) $L_1 \cdot L_2$を次で定義する.

$$L_1 \cdot L_2 = \{ xy \mid x \in L_1, y \in L_2 \}$$

- $L_1 \cdot L_2$ を単に $L_1 L_2$ と書くことがある.

言語の閉包

- 言語 L に対して次のようにおく．

$$L^0 = \{\varepsilon\}$$
$$L^n = L^{n-1} \cdot L \quad (n \geq 1)$$

- さらに，次のようにおく．

$$L^* = \bigcup_{n=0}^{\infty} L^n$$
$$L^+ = \bigcup_{n=1}^{\infty} L^n$$

- L^* を言語 L のKleene閉包(Kleene closure) とよぶ．

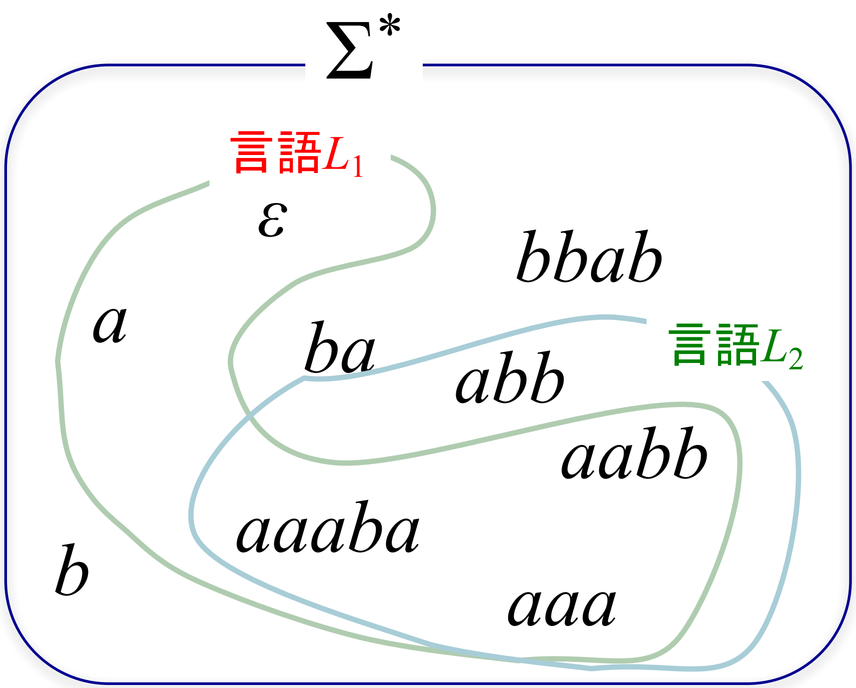
文字列，言語，言語族

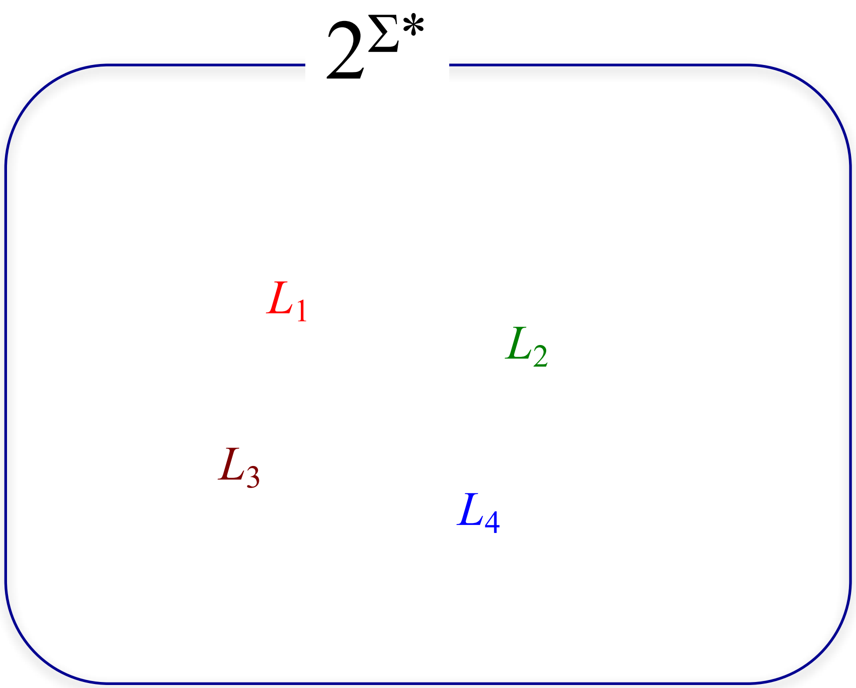
Σ^*： Σ 上の文字列全体の集合

言語： Σ^* の部分集合

言語族：言語の集合

$\Sigma=\{a, b\}$のとき





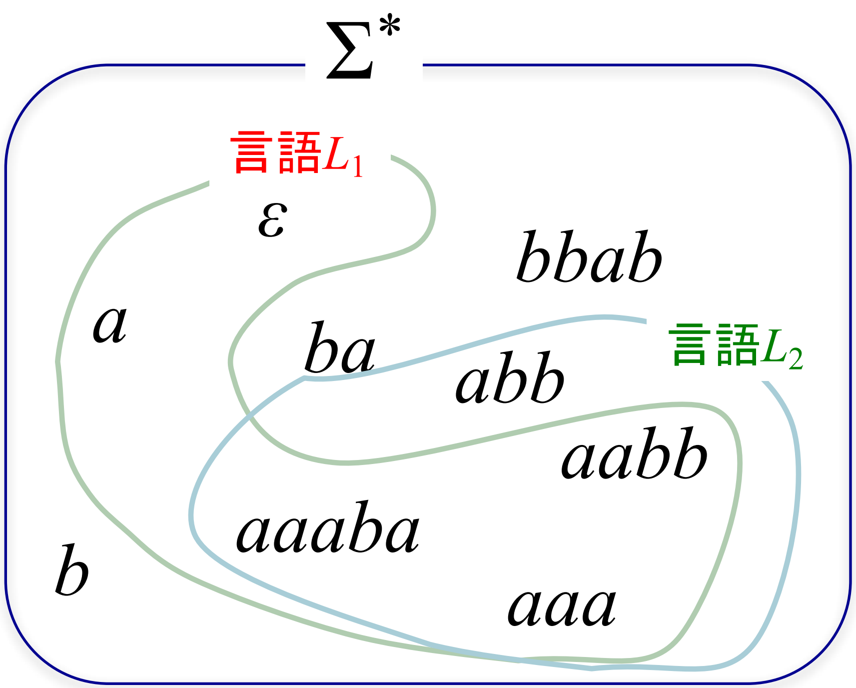
文字列，言語，言語族

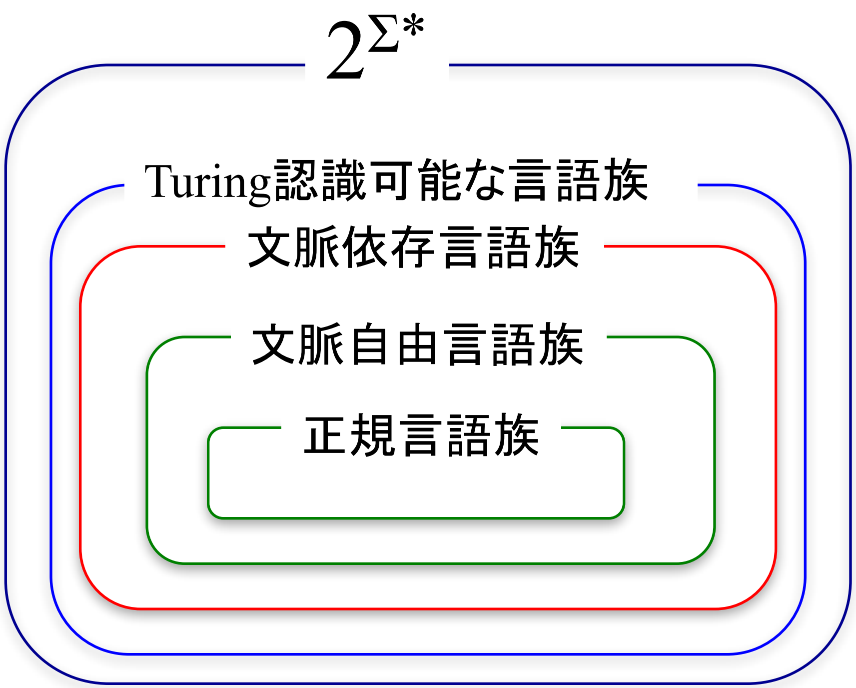
Σ^*：　Σ 上の文字列全体の集合

言語：　Σ^* の部分集合

言語族：言語の集合

$\Sigma=\{a, b\}$のとき

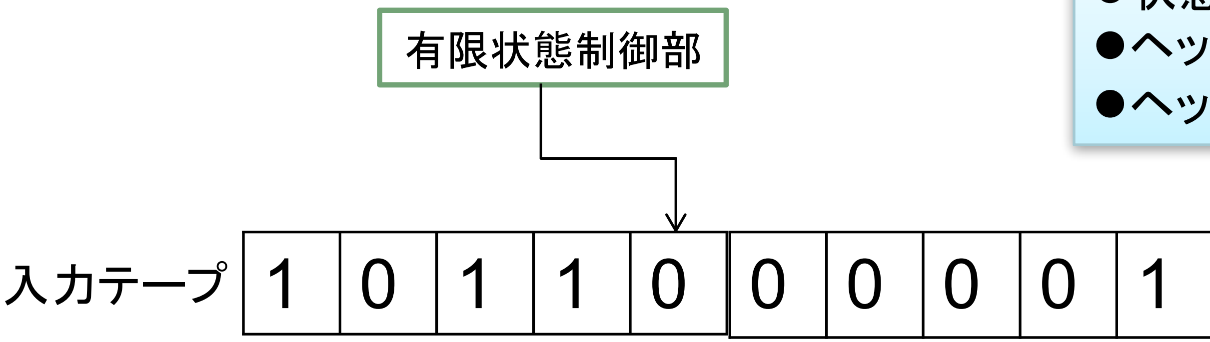




決定性有限オートマトン

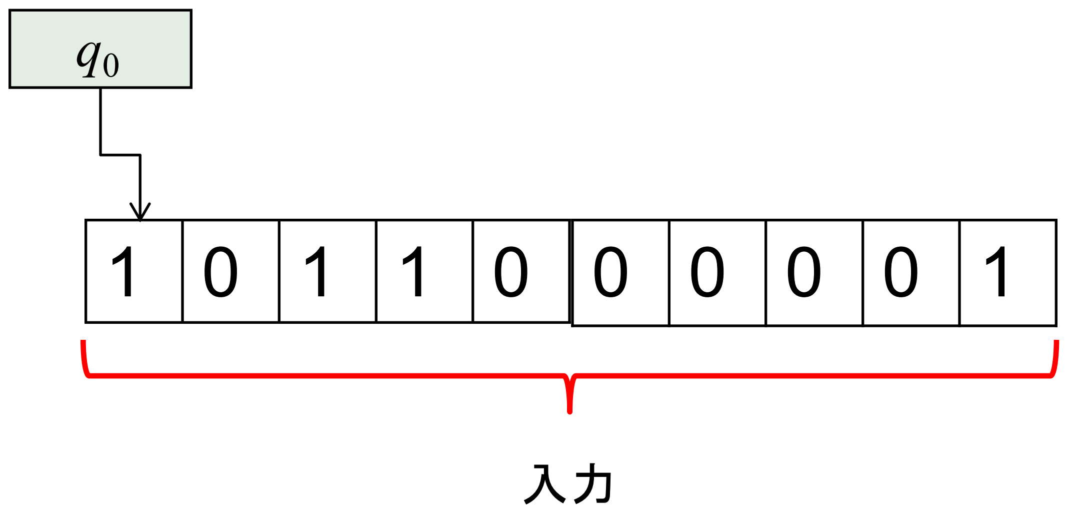
決定性有限オートマトンとは？

- 入力として入力テープに記号列が与えられる.
- テープヘッドは右に1文字ずつ動くことができる.
- 入力記号列を読み切ったとき：
  - 受理状態にあれば，その入力を受理する.
    そうでなければ，受理しない.

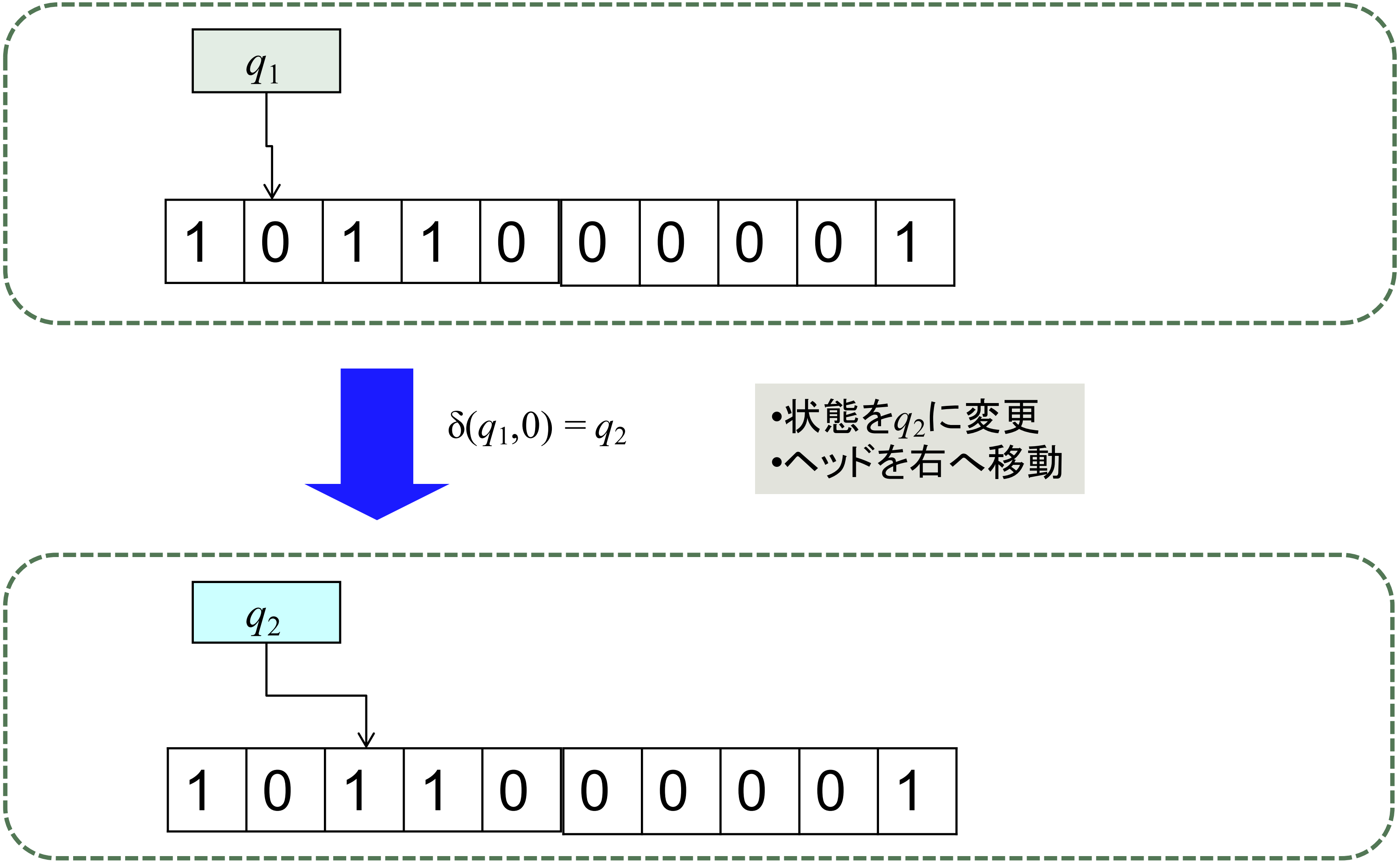


- ●状態の変更
- ●ヘッド位置の文字の読み込み
- ●ヘッドの移動（右へ1コマ分）

計算開始時の決定性有限オートマトン



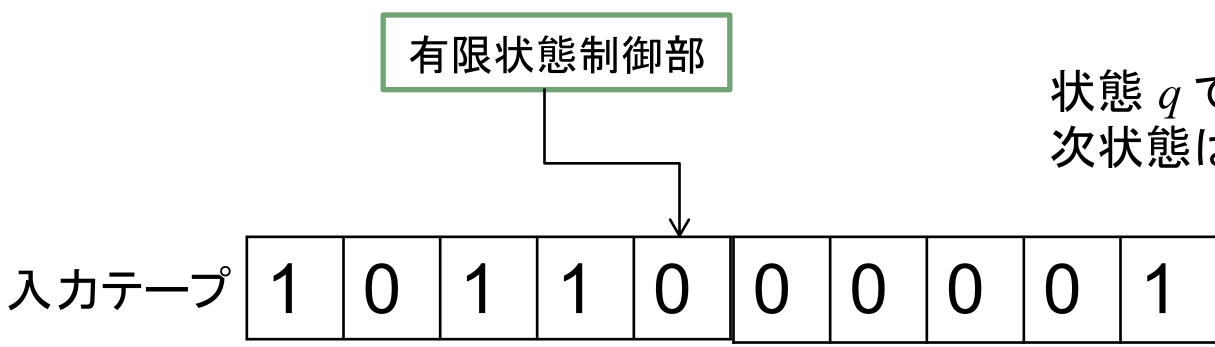
決定性有限オートマトンの動作の1ステップ



決定性有限オートマトン(Deterministic Finite-Automaton; DFA)

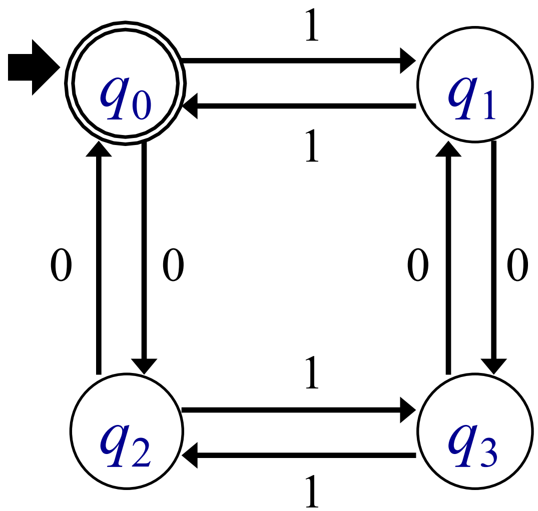
【定義】 決定性有限オートマトンとは，次のような5つ組 $M=(Q, \Sigma, \delta, q_0, F)$ をいう．

- Q は**状態**の空でない有限集合．
- Σ は記号の空でない有限集合で，**アルファベット**と呼ばれる．
- $\delta: Q \times \Sigma \rightarrow Q$ は**状態遷移関数**．
- $q_0 \in Q$ は**開始状態**．
- $F \subseteq Q$ は**最終状態**の集合．



状態 q で入力記号 a を読んだとき，
次状態は $\delta(q, a)$ の値になる

DFAの例



状態集合	$\{ q_0, q_1, q_2, q_3 \}$
アルファベット	$\{ 0, 1 \}$
状態遷移関数	（右表のとおり）
初期状態	q_0
最終状態集合	$\{q_0\}$

状態遷移関数 δ

	0	1
q_0	q_2	q_1
q_1	q_3	q_0
q_2	q_0	q_3
q_3	q_1	q_2

DFAの動作

【定義】 DFA $M=(Q, \Sigma, \delta, q_0, F)$の状態遷移関数 $\delta: Q\times\Sigma\to Q$ に対して，関数 $\hat{\delta}: Q\times\Sigma^*\to Q$ を次のように定める．

(1) 任意の $q\in Q$に対して

$$\hat{\delta}(q, \varepsilon) = q.$$

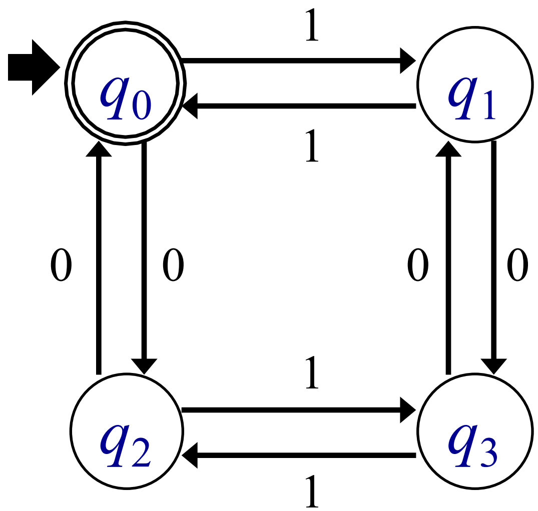
(2) 任意の $w\in\Sigma^*$ と任意の $a\in\Sigma$に対して

$$\hat{\delta}(q, wa) = \delta(\hat{\delta}(q, w), a).$$

$\hat{\delta}(q, x)$:
状態 q からスタートし，文字列 x を左から1文字ずつ読みながら状態遷移を繰り返した際に到達する状態．

以後，関数 $\hat{\delta}$ を δ で表すことにする．

DFAの動作例



入力1000に対する動作例

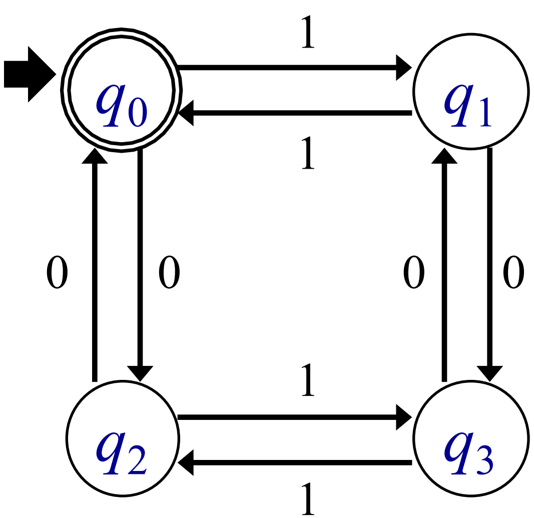
$$q_0 \xrightarrow{1} q_1 \xrightarrow{0} q_3 \xrightarrow{0} q_1 \xrightarrow{0} q_3$$

$$\delta(q_0, 1000) = q_3$$

DFAによる文字列の受理

【定義】 $M = (Q, \Sigma, \delta, q_0, F)$をDFAとする．
M が文字列 $w \in \Sigma^*$ を**受理する**とは，次が成り立つときをいう．

$$\delta(q_0, w) \in F$$



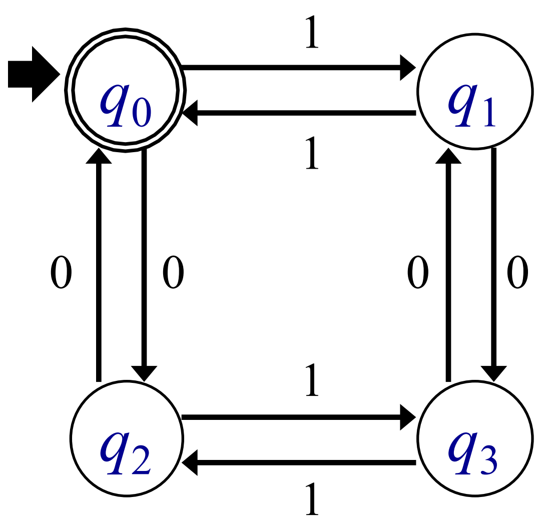
$$\delta(q_0, 11) = q_0 \in F$$

$$\delta(q_0, 1000) = q_3 \notin F$$

DFAの受理する言語

【定義】 DFA $M = (Q, \Sigma, \delta, q_0, F)$に対し，次で定まる言語 $L(M)$ をMの受理する言語と呼ぶ.

$$L(M) = \{\, w \mid w \in \Sigma^*,\ \delta(q_0, w) \in F \,\}$$



$L(M) = \{\, w \mid w \in \{0,1\}^*$, wは 0,1 をそれぞれ偶数個含む$\}$

DFAによる言語の受理

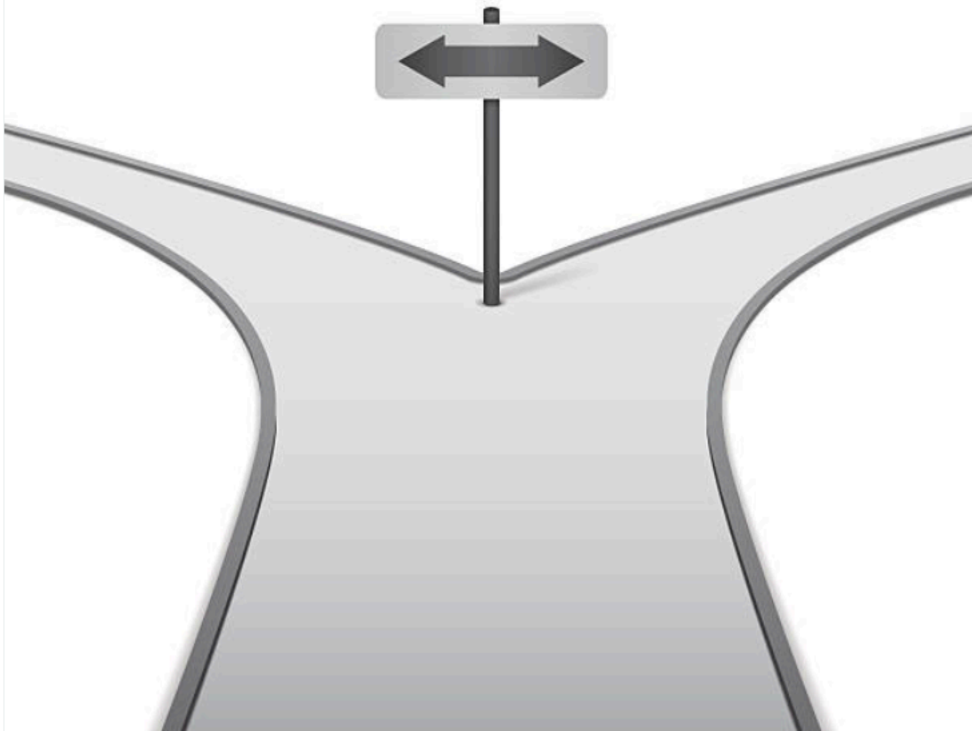
【定義】 DFA M が言語 $L \subseteq \Sigma^*$ を受理するとは，次が成り立つときをいう.

$$L = L(M)$$

非決定性有限オートマトン

「非決定的」な計算とは？

- 次に行うべき動作が一意に定まらないことがある.
- 選択肢に出合う度に適当にいずれかを選んで進む.



非決定性有限オートマトン(Nondeterministic Finite-Automaton; NFA)

【定義】非決定性有限オートマトン(NFA)とは，次を満たす5つ組 $M = (Q, \Sigma, \delta, q_0, F)$ をいう.

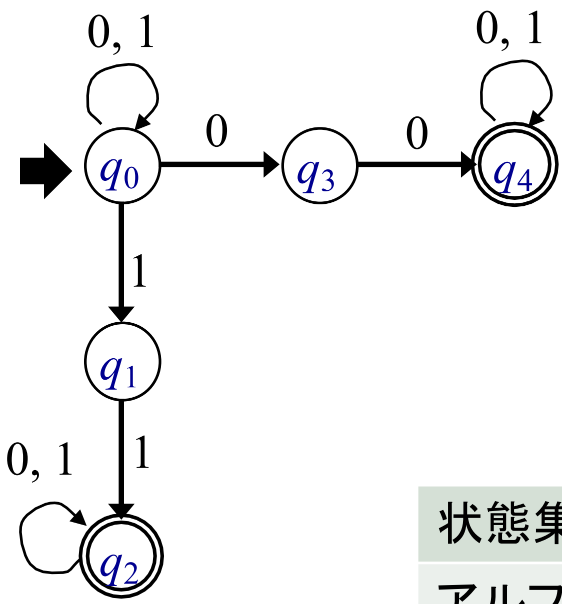
- Q は状態の空でない有限集合.
- Σ は記号の空でない有限集合で，アルファベットと呼ばれる.
- $\delta: Q \times \Sigma \to 2^Q$ は状態遷移関数.
- $q_0 \in Q$ は開始状態.
- $F \subseteq Q$ は最終状態の集合.

$\delta(q, a) \in 2^Q$，すなわち，$\delta(q, a) \subseteq Q$ である.

状態 q で入力記号 a を読んだとき，
次状態は 集合 $\delta(q, a)$ の要素のいずれかとなる.

DFA は NFA の特別な場合 (つねに$|\ \delta(q, a)\ |=1$) とみなせる.

NFAの例



状態集合	$\{ q_0, q_1, q_2, q_3, q_4 \}$
アルファベット	$\{ 0, 1 \}$
状態遷移関数	（右表のとおり）
初期状態	q_0
最終状態集合	$\{q_2, q_4\}$

状態遷移関数 δ

	0	1
q_0	$\{q_0, q_3\}$	$\{q_0, q_1\}$
q_1	$\varnothing$	$\{q_2\}$
q_2	$\{q_2\}$	$\{q_2\}$
q_3	$\{q_4\}$	$\varnothing$
q_4	$\{q_4\}$	$\{q_4\}$

NFAの動作の定義

【定義】 NFA $M=(Q, \Sigma, \delta, q_0, F)$の状態遷移関数 $\delta: Q\times\Sigma \rightarrow 2^Q$ に対して，関数 $\hat{\delta}: Q\times\Sigma^* \rightarrow 2^Q$ を次のように定める.

(1) 任意の $q\in Q$に対して $\hat{\delta}(q, \varepsilon) = \{q\}$.

(2) 任意の $w\in\Sigma^*$ と任意の $a \in \Sigma$に対して

$$\hat{\delta}(q, wa) = \bigcup_{r\in \hat{\delta}(q, w)} \delta(r, a).$$

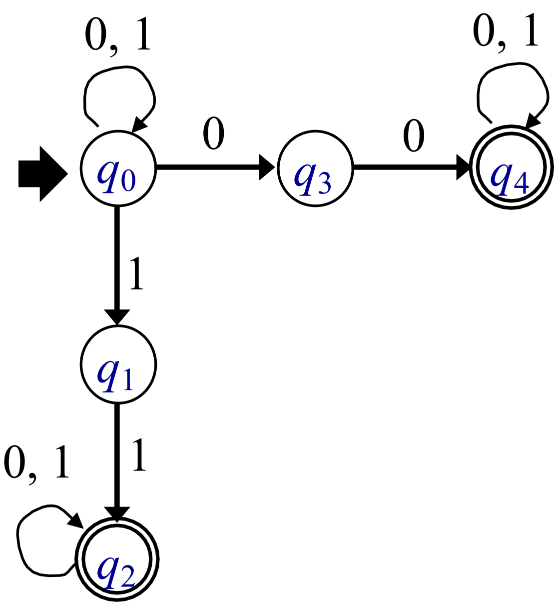
さらに，関数 $\hat{\delta}: 2^Q\times\Sigma^* \rightarrow 2^Q$ へ拡張する.

(3) 任意の$P\subseteq Q$と任意の $x\in\Sigma^*$に対して $\hat{\delta}(P, x) = \bigcup_{r\in P} \delta(r, x)$.

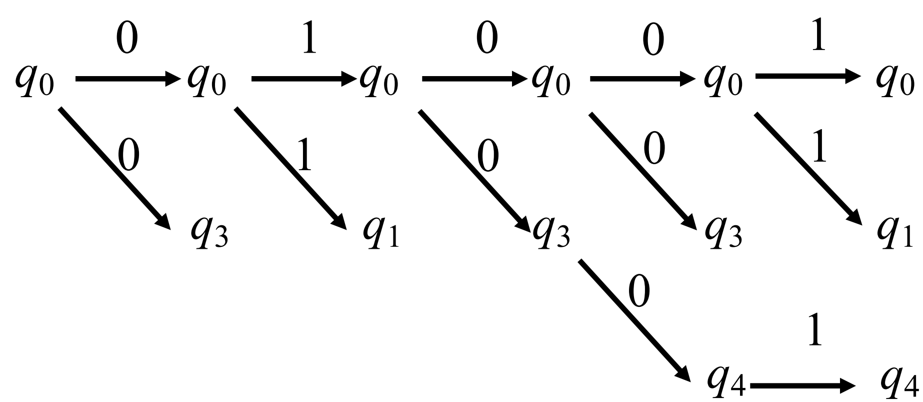
$\hat{\delta}(q, x)$:
状態 q からスタートし，文字列 x を左から1文字ずつ読みながら状態遷移を繰り返した際に到達可能な状態の集合.

以後，関数 $\hat{\delta}$ を δ で表すことにする.

NFAの動作例



入力01001に対する動作例

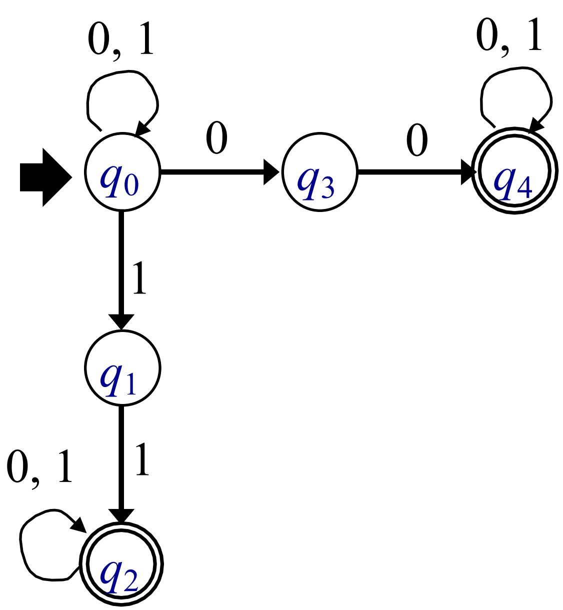


$$\delta(q_0, 01) = \{q_0, q_1\}$$
$$\delta(q_0, 01001) = \{q_0, q_1, q_4\}$$

NFAによる文字列の受理

【定義】 $M = (Q, \Sigma, \delta, q_0, F)$をNFAとする．
M が文字列 $w \in \Sigma^*$ を受理するとは，次が成り立つときをいう．

$$\delta(q_0, w) \cap F \neq \varnothing$$



$$\delta(q_0, 01001) = \{q_0, q_1, q_4\}$$

$$\delta(q_0, 01001) \cap F = \{q_4\} \neq \varnothing$$

NFAの受理する言語

【定義】 NFA $M = (Q, \Sigma, \delta, q_0, F)$に対し，次で定まる言語 $L(M)$ をMの受理する言語と呼ぶ.

$$L(M) = \{\, w \mid w \in \Sigma^*,\ \delta(q_0, w) \cap F \neq \varnothing \,\}$$

NFAとDFAの等価性

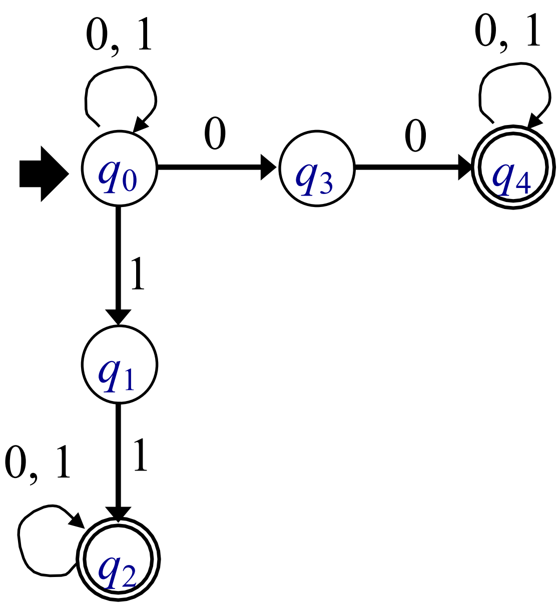
【定理2.1】L を任意の言語とする．このとき，次の(1)(2)は等価である．

(1) あるNFA M が存在して $L=L(M)$ となる．

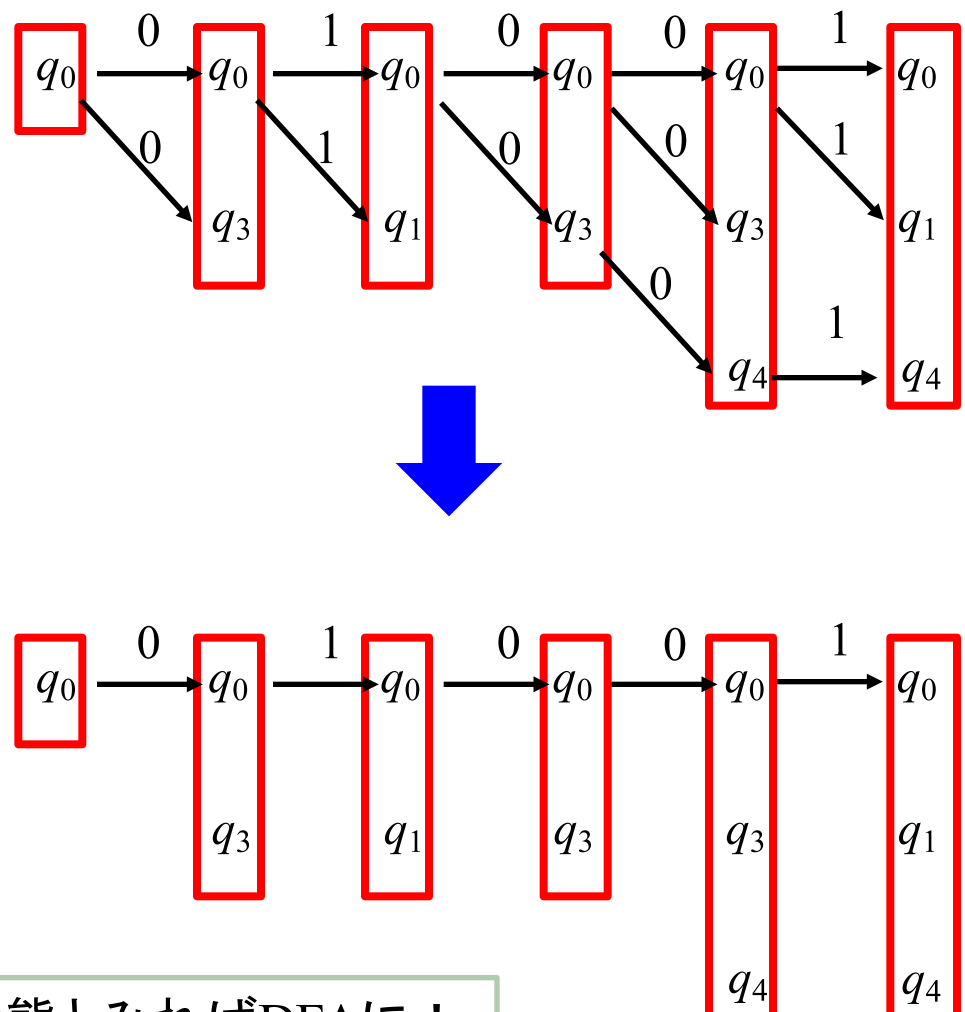
(2) あるDFA M が存在して $L=L(M)$ となる．

DFAはNFAの特別な場合とみなせるため，(2) $\Rightarrow$ (1)は明らか．
よって，(1) $\Rightarrow$ (2)を示せばよい．

「(1) ⇒ (2)」の証明のアイデア



入力01001に対する動作例



NFAの状態の集合を状態とみればDFAに！

定理2.1の証明

- (2) $\Rightarrow$(1)は自明. (1) $\Rightarrow$(2)を示す.
- NFA $M = (Q, \Sigma, \delta, q_0, F)$ に対して
  DFA $M' = (Q', \Sigma, \delta', q_0', F')$ を次で定義する.
  - $Q' = 2^Q$.
  - $\delta': Q' \times \Sigma \to Q'$ は次で定まる状態遷移関数.
    $$\delta'(P, a) = \bigcup_{q \in P} \delta(q, a).$$
  - $q_0' = \{q_0\}$.
  - $F' = \{ P \mid P \in Q', P \cap F \neq \varnothing \}$.
- 次の事実1, 2より, $L(M) = L(M')$ が成り立つ.
  [事実1] すべての $x \in \Sigma^*$ に対して $\delta(q_0, x) = \delta'(q_0', x)$.
  [事実2] すべての $x \in \Sigma^*$ に対して
  $$\delta(q_0, x) \cap F \neq \varnothing \Leftrightarrow \delta'(q_0', x) \in F'.$$

QED

事実1の証明

- すべての $x \in \Sigma^*$ に対して $\delta(q_0, x) = \delta'(q_0', x)$であることを$|x|$に関する帰納法により示す.
- $|x|=0$のとき, $\delta(q_0, \varepsilon)=\{q_0\}$, $\delta'(q_0', \varepsilon) = q_0'=\{q_0\}$より成立.
- $|x|\geq 1$のとき. $x=wa$ $(w\in\Sigma^*, a \in \Sigma)$ とおく.
  - $P=\delta'(q_0', w)$とおくと,
    $\delta'(q_0', x) = \delta'(q_0', wa) = \delta'(\delta'(q_0', w), a) = \delta'(P, a)$.
  - 帰納法の仮定より, $\delta(q_0, w) = \delta'(q_0', w)$. よって,
    $\delta(q_0, x) = \delta(q_0, wa) = \delta(\delta(q_0, w), a) = \delta(P, a)$.
  - δ'の定義より $\delta'(P, a) = \bigcup_{q\in P}\delta(q, a) = \delta(P, a)$.
  - よって, $\delta'(q_0', x) = \delta'(P, a) = \delta(P, a) = \delta(q_0, x)$.

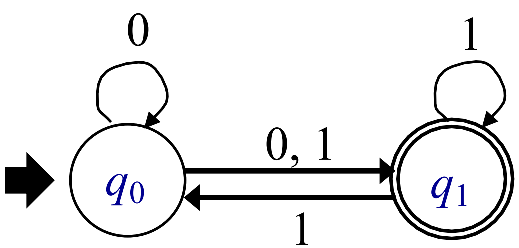
QED

事実2の証明

- すべての $x \in \Sigma^*$ に対して
  $$\delta(q_0, x) \cap F \neq \varnothing \Leftrightarrow \delta'(q_0', x) \in F'$$
  であることを示す.
- $F' = \{ P \mid P \in Q', P \cap F \neq \varnothing \}$であるから,
  $\delta'(q_0', x) \in F' \Leftrightarrow \delta'(q_0', x) \cap F \neq \varnothing$
- 事実1より $\delta'(q_0', x) = \delta(q_0, x)$であるから,
  $\delta'(q_0', x) \cap F \neq \varnothing \Leftrightarrow \delta(q_0, x) \cap F \neq \varnothing$

QED

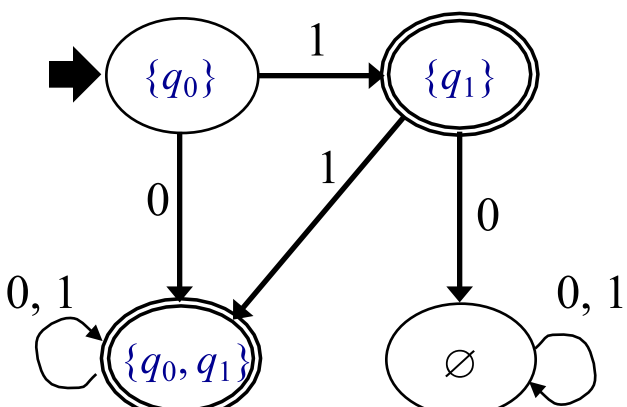
例



状態集合	$\{ q_0, q_1 \}$
アルファベット	$\{ 0, 1 \}$
状態遷移関数	（右表のとおり）
初期状態	q_0
最終状態集合	$\{q_1\}$

	0	1
q_0	$\{q_0, q_1\}$	$\{q_1\}$
q_1	$\varnothing$	$\{q_0, q_1\}$

NFAをDFAに変換



状態集合	$\{ \varnothing, \{q_0\}, \{q_1\}, \{ q_0, q_1\}\}$
アルファベット	$\{ 0, 1 \}$
状態遷移関数	（右表のとおり）
初期状態	$\{q_0\}$
最終状態集合	$\{\{q_1\}, \{q_0, q_1\}\}$

	0	1
$\varnothing$	$\varnothing$	$\varnothing$
$\{q_0\}$	$\{q_0, q_1\}$	$\{q_1\}$
$\{q_1\}$	$\varnothing$	$\{q_0, q_1\}$
$\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_0, q_1\}$

拡張されたNFA

拡張されたNFA

- 教科書によっては，NFAについて以下の定義を採用するものもある．

【定義】非決定性有限オートマトン(NFA)とは，次を満たす5つ組 $M = (Q, \Sigma, \delta, Q_0, F)$ をいう．

- Q は状態の空でない有限集合．
- Σ は記号の空でない有限集合で，アルファベットと呼ばれる．
- $\delta: Q \times \Sigma \to 2^Q$ は状態遷移関数．
- $Q_0 \subseteq Q$ は開始状態の集合．
- $F \subseteq Q$ は最終状態の集合．

計算開始時には，Q_0の要素のいずれかが初期状態となる．

拡張されたNFAによる受理

- 拡張されたNFA $M = (Q, \Sigma, \delta, Q_0, F)$ が文字列 $x \in \Sigma^*$を受理するとは，次が成り立つときをいう.

$$\delta(Q_0, x) \cap F \neq \varnothing$$

拡張されたNFAの能力

- 拡張されたNFAについても，定理2.1と同様の命題が成立する.
- 証明はほぼ同じである. 実際，定理2.1の証明においてq0'= Q0とすればよい.

NFAの定義として，どちらの定義を用いても本質的な差異はない.

おまけ：拡張されたNFAからNFAへの変換

- 拡張されたNFAを $M = (Q, \Sigma, \delta, Q_0, F)$とする.
- NFA $M' = (Q \cup \{q_0'\}, \Sigma, \delta', q_0', F')$ を次で定める.
  - $q_0' \notin Q$.
  - $\delta'(q, a) = \begin{cases} \delta(q, a), & \text{if } (q, a) \in Q \times \Sigma \\ \bigcup_{r \in Q_0} \delta(r, a), & \text{if } (q, a) \in \{q_0'\} \times \Sigma \end{cases}$
  - $F' = \begin{cases} F, & \text{if } Q_0 \cap F = \emptyset \\ F \cup \{q_0'\}, & \text{otherwise} \end{cases}$

例