有限オートマトン

離散数学・オートマトン
2024 年後期
佐賀大学理工学部　只木進一

オートマトンと形式言語: Automata and Formal Languages

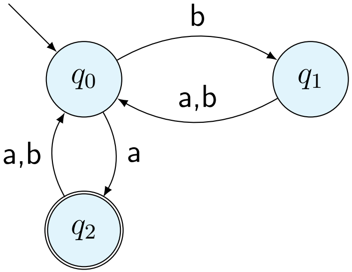
- オートマトン (Automaton)
  - 計算の抽象モデル
  - テープからの入力による状態遷移
  - 「計算する」とは何かを考える
  - automata は複数形
- 形式言語 (Formal Language)
  - オートマトンが受理する言語
    入力を正しく処理できるか
  - 文法を数学的に分析

決定性有限オートマトン
Deterministic Finite State Automata: DFA

$$M = \langle Q, \Sigma, \delta, q_0, F \rangle \tag{2.1}$$

- Q: 内部状態の有限集合
- Σ: 入力アルファベット、つまり入力記号の有限集合
- $\delta : Q \times \Sigma \to Q$: 状態遷移関数
    - ある状態 q で文字 a を読むと、状態が p に遷移する
    - $\delta(q, \mathsf{a}) = p$
- $q_0 \in Q$: 初期状態
- $F \subseteq Q$: 受理状態の集合
    - $q \in F$ に到達する入力を受理する

例 2.1:



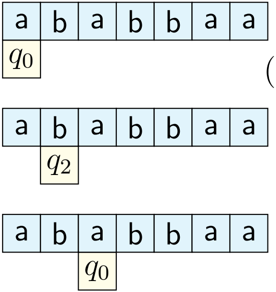
$$Q = \{q_0, q_1, q_2\}$$
$$\Sigma = \{\mathsf{a}, \mathsf{b}\}$$
$$F = \{q_2\}$$

遷移関数 δ

	a	b
q_0	q_2	q_1
q_1	q_0	q_0
q_2	q_0	q_0

動作イメージ

テープヘッドが移動して、テープ上の文字を読み取る。



$$(q_0, \mathsf{ababbaa}) \vdash_M (q_2, \mathsf{babbaa})$$

$$(q_2, \mathsf{babbaa}) \vdash_M (q_0, \mathsf{abbaa})$$

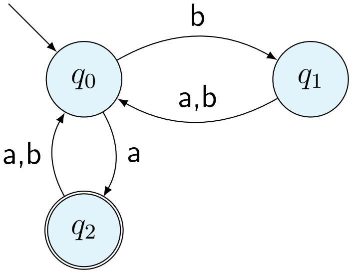
$$
\begin{aligned}
(q_0, \mathsf{ababbaa}) &\vdash_M (q_2, \mathsf{babbaa}) \\
&\vdash_M (q_0, \mathsf{abbaa}) \\
&\vdash_M (q_2, \mathsf{bbaa}) \\
&\vdash_M (q_0, \mathsf{baa}) \\
&\vdash_M (q_1, \mathsf{aa}) \\
&\vdash_M (q_0, \mathsf{a}) \\
&\vdash_M (q_2, \epsilon)
\end{aligned}
$$

遷移関数 δ

	a	b
q_0	q_2	q_1
q_1	q_0	q_0
q_2	q_0	q_0

ϵは長さ 0 の文字列を表す

例:2.1への入力 bbaba



$\vdash_M$ の推移的閉包と受理言語

- 入力 $w \in \Sigma^*$(Σ^* は Σ の要素の 0 個以上の列) によって、初期状態 q_0 から状態 q へ遷移し、テープに残っている文字列が w'

$$(q_0, w) \vdash_M^* (q, w') \tag{3.1}$$

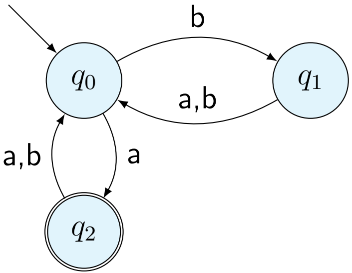
- 入力 w を受理

$$(q_0, w) \vdash_M^* (q_F, \epsilon), \quad q_F \in F \tag{3.2}$$

- 受理言語

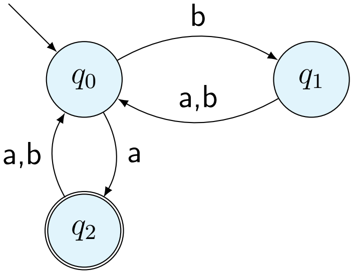
$$L(M) = \{w \in \Sigma^* \mid (q_0, w) \vdash_M^* (q_F, \epsilon), \quad q_F \in F\} \tag{3.3}$$

例:2.1 の場合



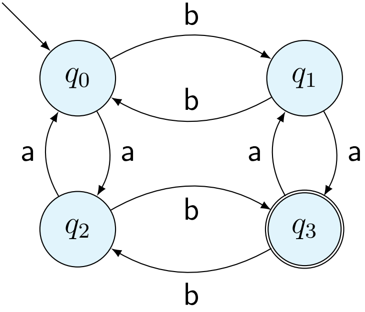
$$
\begin{aligned}
(q_0, \mathsf{aaaba}) &\vdash (q_2, \mathsf{aaba}) \vdash (q_0, \mathsf{aba}) \\
&\vdash (q_2, \mathsf{ba}) \vdash (q_0, \mathsf{a}) \vdash (q_2, \epsilon) \\
(q_0, \mathsf{babaa}) &\vdash (q_1, \mathsf{abaa}) \vdash (q_0, \mathsf{baa}) \\
&\vdash (q_1, \mathsf{aa}) \vdash (q_0, \mathsf{a}) \vdash (q_2, \epsilon)
\end{aligned}
$$

受理する入力の例



a, aaa, aba, baa, bba,
aaaaa, aaaba, abaaa,
babaa, babba, bbbaa,
bbbba

例 3.1:

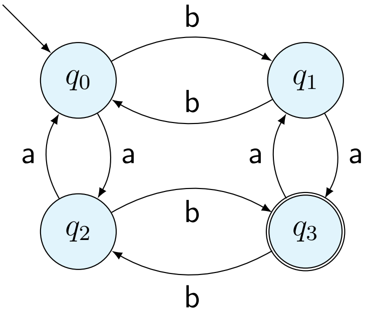


$$Q = \{q_0, q_1, q_2, q_3\}$$
$$\Sigma = \{\mathsf{a}, \mathsf{b}\}$$
$$F = \{q_3\}$$

遷移関数 δ

	a	b
q_0	q_2	q_1
q_1	q_3	q_0
q_2	q_0	q_3
q_3	q_1	q_2

例 3.1: 動作例

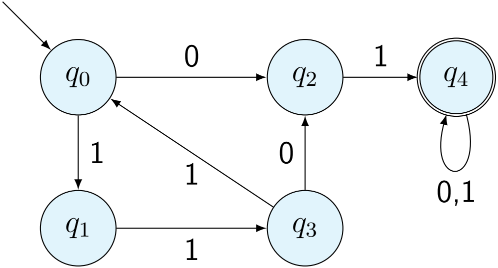


$$(q_0, \mathsf{aaaaab}) \vdash (q_2, \mathsf{aaaab}) \vdash (q_0, \mathsf{aaab}) \vdash (q_2, \mathsf{aab})$$
$$\vdash (q_0, \mathsf{ab}) \vdash (q_2, \mathsf{b}) \vdash (q_3, \epsilon)$$
$$(q_0, \mathsf{abbaba}) \vdash (q_2, \mathsf{bbaba}) \vdash (q_3, \mathsf{baba}) \vdash (q_2, \mathsf{aba})$$
$$\vdash (q_0, \mathsf{ba}) \vdash (q_1, \mathsf{a}) \vdash (q_3, \epsilon)$$

例 3.1: 受理する文字列例 (長さ 5 まで)

ab ba aaab aaba abaa abbb baaa babb bbab bbba

例 3.2:



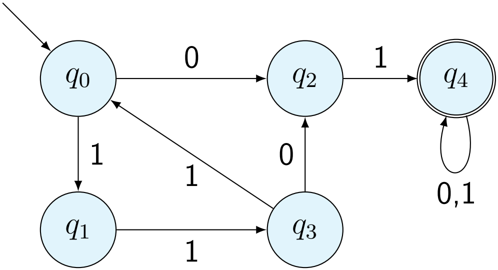
$Q = \{q_0, q_1, q_2, q_3, q_4\}$
$\Sigma = \{0, 1\}$
$F = \{q_4\}$

遷移関数 δ

	0	1
q_0	q_2	q_1
q_1		q_3
q_2		q_4
q_3	q_2	q_0
q_4	q_4	q_4

空欄に注意

例 3.2: 動作例



$$
\begin{aligned}
(q_0, 1110101) &\vdash (q_1, 110101) \vdash (q_3, 10101) \vdash (q_0, 0101) \\
&\vdash (q_2, 101) \vdash (q_4, 01) \vdash (q_4, 1) \vdash (q_4, \epsilon) \\
(q_0, 1101010) &\vdash (q_1, 101010) \vdash (q_3, 01010) \vdash (q_2, 1010) \\
&\vdash (q_4, 010) \vdash (q_4, 10) \vdash (q_4, 0) \vdash (q_4, \epsilon)
\end{aligned}
$$

例 3.2: 受理する文字列例 (長さ 5 まで)

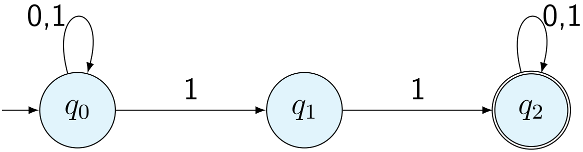
01, 010, 011, 0100, 0101, 0110, 0111, 1101, 01000, 01001, 01010, 01011, 01100, 01101, 01110, 01111, 11010, 11011, 11101

非決定性有限オートマトン
Non-deterministic Finite State Automata: NFA

$$M = \langle Q, \Sigma, \delta, q_0, F \rangle \tag{4.1}$$

- Q: 内部状態の集合
- Σ: 入力アルファベット
- $\delta : Q \times \Sigma \to 2^Q$: 状態遷移関数。
  2^Q は、Q のべき集合、つまり Q の部分集合の族。遷移先が複数であることを許容することに注意。
- $q_0 \in Q$: 初期状態
- $F \subseteq Q$: 受理状態

例 4.1:

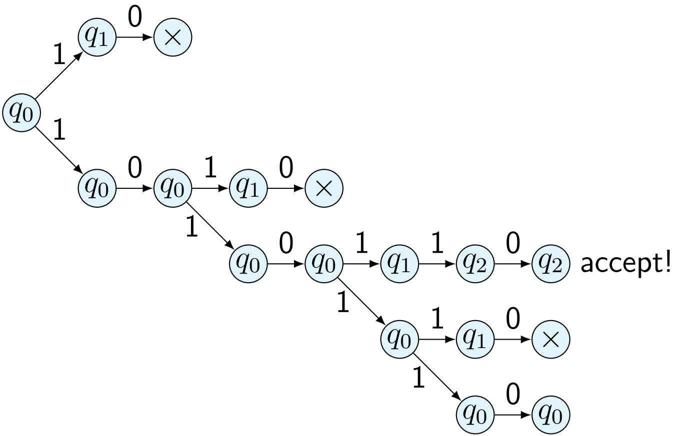


$$Q = \{q_0, q_1, q_2\}, \quad \Sigma = \{0, 1\}, \quad F = \{q_2\}$$

遷移関数 δ

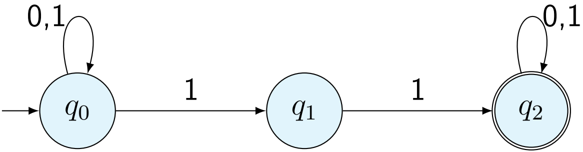
	0	1
q_0	$\{q_0\}$	$\{q_0, q_1\}$
q_1	$\emptyset$	$\{q_2\}$
q_2	$\{q_2\}$	$\{q_2\}$

動作例: 入力 1010110



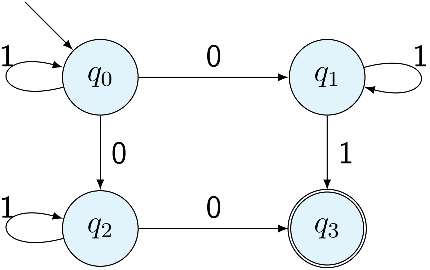
入力が引き起こす状態遷移のうちで、受理状態に至る場合が一つでもあれば、その入力を受理する。

長さ5以下の受理入力



11, 011, 110, 111, 0011, 0110, 0111, 1011, 1100, 1110, 1111, 00011, 00110, 00111, 01011, 01100, 01110, 01111, 10011, 10110, 10111, 11000, 11011, 11100, 11110, 11111

例 4.2:



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

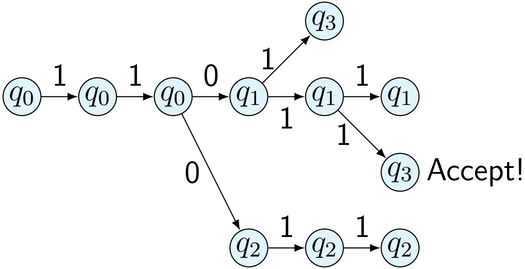
遷移関数 δ

δ	0	1
q_0	$\{q_1, q_2\}$	$\{q_0\}$
q_1	$\emptyset$	$\{q_1, q_3\}$
q_2	$\{q_3\}$	$\{q_2\}$
q_3	$\emptyset$	$\emptyset$

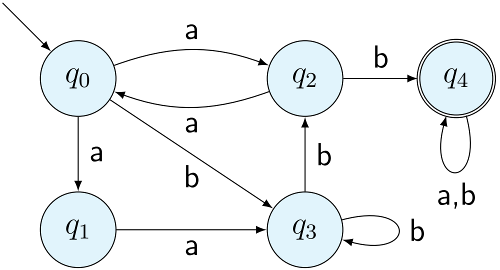
長さ5以下の受理入力

00, 01, 010, 011, 100, 101, 0110, 0111, 1010, 1011, 1100, 1101, 01110, 01111, 10110, 10111, 11010, 11011, 11100, 11101

動作例: 入力 11011



例 4.3:

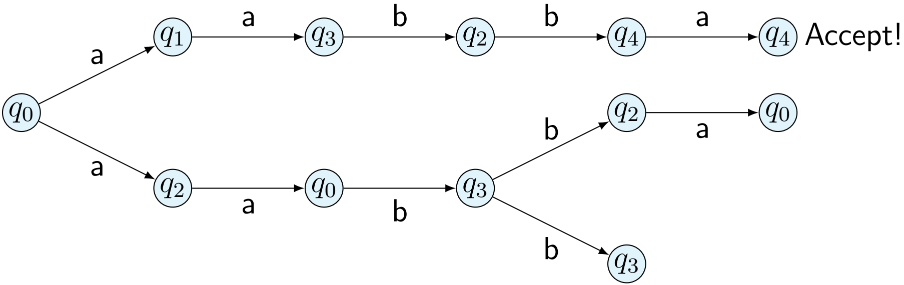


$$Q = \{q_0, q_1, q_2, q_3, q_4\}, \Sigma = \{\mathsf{a}, \mathsf{b}\}, F = \{q_4\}$$

遷移関数 δ

	a	b
q_0	$\{q_1, q_2\}$	$\{q_3\}$
q_1	$\{q_3\}$	$\emptyset$
q_2	$\emptyset$	$\{q_4\}$
q_3	$\emptyset$	$\{q_2, q_3\}$
q_4	$\{q_4\}$	$\{q_4\}$

動作例: 入力 aabba



疑問

- オートマトンが受理する文字列の集合を記述する方法
  - 文字列パターンを記述する方法
- NFA と DFA は本質的に異なるのか
  - 受理する文字列集合は異なるのか
  - 能力は異なるか