7　形式言語論 (1) – 有限オートマトン –

形式言語の研究は, 計算機科学の根幹をなす. 現在の計算機科学の基礎は, その上の自然言語の文法を数学的に定式化するところから出発した. 計算機科学での大きなテーマは「計算機の本質的な能力とその限界とはなにか」という疑問への回答を与えることである. ここからは, これらの数学的な基礎となる部分を学ぼう.

• 7-1 : 有限オートマトンの導入

有限オートマトンはメモリが著しく制限されているような計算機に対する基本モデルである. 著しく制限されているとはいえ, 実は多くの有益な仕事ができている.

例 7-1　次のような単純なモデルを考える. 一方向にしか進めない（入口専用, あるいは出口専用）の自動ドアの制御部のシステムを考えよう. 自動ドアは, 入る側にある入口センサーと通過した先に通過センサーをもち, 自動ドアの制御部は,「OPEN」か「CLOSED」という自動ドアの 2 つの状態のどちらかをとる. この制御部には, 次の 4 パターンの条件のいずれかが入力される.

(A) どちらのセンサーでも人を検知していない.
(B) 入口センサーでのみ人を検知した.
(C) 通過センサーでのみ人を検知した.
(D) 入口センサーと通貨センサーの両方で人を検知した.

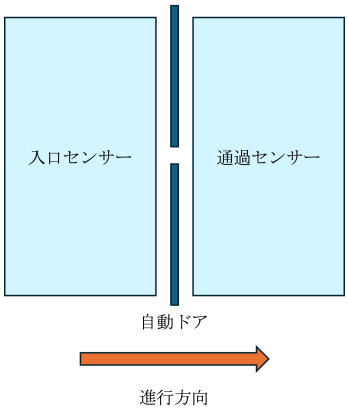


図 1　自動ドアの単純モデル

すると, 自動ドアの制御部は受け付けた入力によって, 次の表に従いある状態からある状態へと動く.

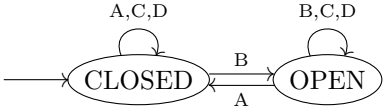
	A	B	C	D
CLOSED	CLOSED	OPEN	CLOSED	CLOSED
OPEN	CLOSED	OPEN	OPEN	OPEN

表 1　自動ドアの状態

例えば, CLOSED の状態から始動して, 制御部が B, C, A, B, D, C, A という順で入力を受け付けたとする. このとき, 自動ドアの状態は

$$\text{CLOSED} \xrightarrow{B} \text{OPEN} \xrightarrow{C} \text{OPEN} \xrightarrow{A} \text{CLOSED} \xrightarrow{B} \text{OPEN} \xrightarrow{D} \text{OPEN} \xrightarrow{C} \text{OPEN} \xrightarrow{A} \text{CLOSED}$$

このような状態の遷移を, **状態遷移図** と呼ばれるかたちで図示することができる.



• 7-2 : 決定性有限オートマトンの厳密な定義

では, これを定式化して（決定性）有限オートマトンを定義する. 状態遷移図は, 人間には直感的には理解しやすいが, 以下の 2 つの理由から厳密な定義が必要となる.

- 厳密な定義は, 有限オートマトンには何が許されているのかについて不明確な点をすべて解決する. 例えば, 受理状態が 1 つもないことを許すのか, 各状態から出ていく遷移は, 取りうる入力文字についてただ 1 つ定まらなければならないのか, などである.
- 定義は, 有限オートマトンの正確な記法を与える. これは, 有限オートマトンの細部にまで正確に記述されており, 不明瞭さを排除する.

では, 有限オートマトンには何が必要か考えよう. まず, 「状態」とよばれる集合が必要である. 第二に, 入力となる文字列, つまり「入力につかう文字」の集合が必要である. 第三に, 今の状態と受け取った文字に対応して次の状態へ遷移する「動作規則」が必要である. 第四に, どの状態から出発するのかという「開始状態」が必要である. 最後に, 何を受理して何を拒否するのかという「受理する状態のリスト」が必要である.

まずは「文字」について厳密に定義しよう.

定義 7.1. **アルファベット** とは, 空でない有限集合である. 以下, Σ をアルファベットとする.

(a) Σ の元を **文字** といい, 文字を並べた有限列を **文字列** と呼ぶ. 文字列全体からなる集合を Σ^* で表す.
(b) 文字列 $w \in \Sigma^*$ に含まれる文字の数をその **長さ** と呼び, これを $\ell(w)$ で表す. 長さが 0 の文字列を **空列** と呼び, 特別な記号 ε で表す. 空列ではない文字列全体を $\Sigma^+ := \Sigma^* \setminus \{\varepsilon\}$ で表す.
(c) Σ^* の部分集合を **言語** という.

ここでは, アルファベットの文字をタイプライタフォントで表すことにする.

例 7-2　(1) アルファベット $\Sigma = \{\texttt{a},\texttt{b},\texttt{c},\ldots,\texttt{y},\texttt{z}\}$ とする. このとき, $w = \texttt{risansuugaku}$ は Σ 上の文字列であり, $\ell(w) = 12$ である.
(2) $\Sigma = \{\texttt{0},\texttt{1}\}$ であるとき, $\Sigma^* = \{\varepsilon, \texttt{0}, \texttt{1}, \texttt{00}, \texttt{01}, \texttt{11}, \texttt{000}, \ldots\}$ となり, $\Sigma^+ = \{\texttt{0}, \texttt{1}, \texttt{00}, \texttt{01}, \texttt{11}, \texttt{000}, \ldots\}$ である.

では, （決定性）有限オートマトンを厳密に定義しよう.

定義 7.2. 2 つの有限集合 K, Σ と写像 $\delta : K \times \Sigma \to K$, および $q_0 \in K$ と $F \subset K$ の 5 つ組 $M = (K, \Sigma, \delta, q_0, F)$ を **決定性有限オートマトン** と呼ぶ. ここで,

- K の元を **状態**,
- Σ の元を **入力アルファベット**,
- $\delta : K \times \Sigma \longrightarrow K$ を **遷移関数**,
- $q_0 \in K$ を **開始状態**,
- $F \subset K$ の元を **受理状態**

と呼ぶ.

決定性有限オートマトン $M = (K, \Sigma, \delta, q_0, F)$ のそれぞれが具体的に与えられたとき, それを状態遷移図を使って視覚的に明示することができる.

例 7-3　集合 K, Σ, F を

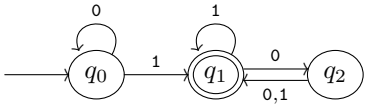
$$K = \{q_0, q_1, q_2\}, \quad \Sigma = \{\texttt{0}, \texttt{1}\}, \quad F = \{q_1\}$$

として, 遷移関数 $\delta : K \times \Sigma \to K$ を以下のように与える.

δ	0	1
$\to q_0$	q_0	q_1
$^*q_1$	q_2	q_1
q_2	q_1	q_1

ただし, 表の各行は状態を, 各列は入力アルファベットに対応していおり, $\to q_0$ は q_0 が開始記号であること, $^*q_1$ は q_1 が受理状態であることを表している. このとき, 決定性有限オートマトン $M = (K, \Sigma, \delta, q_0, F)$ は ① のように図示される.

M :



..... ①

ここで, q_0 に入ってくる矢印は, q_0 が開始記号であることを表し, q_1 が二重丸で囲まれているのは, q_1 が受理状態であることを表している. M に文字列 1100 を入力すると, M は以下のように動作する.

$$q_0 \xrightarrow{1} q_1 \xrightarrow{1} q_1 \xrightarrow{0} q_2 \xrightarrow{0} q_1$$

入力の読み込みが終了したとき, 状態は q_1 で受理状態であるから, M は文字列 1100 を受理する.

• 7-2 : 有限オートマトンが認識する言語

決定性有限オートマトン M があるとき, 文字列を入力すると M が動作し, 状態が遷移していく. 読み込み終了後に, 受理状態であれば, M はその文字を受理する. そのような, M が受理する文字列全体からなる言語を得ることができるが, これを厳密に定めよう. そのために, M の遷移関数 δ をその入力として「状態」と「文字列」が入力できる $\widehat{\delta}$ に拡張する.

定義 7.3. 決定性有限オートマトン $M = (K, \Sigma, \delta, q_0, F)$ に対して, **拡張された遷移関数** $\widehat{\delta} : K \times \Sigma^* \to K$ を $(q, w) \in K \times \Sigma^*$ に対して次のように帰納的に定義する.

(i) $w = \varepsilon$ のとき, $\widehat{\delta}(q, \varepsilon) = q$.
(ii) $w = xa$ $(x \in \Sigma^*,\ a \in \Sigma)$ のとき, $\widehat{\delta}(q, w) = \delta\left(\widehat{\delta}(q, x), a\right)$. 特に, $a \in \Sigma$ に対して. $\widehat{\delta}(q, a) = \delta(q, a)$.

例 7-4　**例 7-3** で与えられている M に文字列 1100 を入力すると,

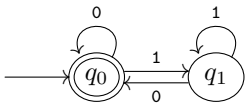
- $\widehat{\delta}(q_0, 1) = q_1$,
- $\widehat{\delta}(q_0, 11) = \delta(q_1, 1) = q_1$,
- $\widehat{\delta}(q_0, 110) = \delta(q_1, 0) = q_2$,
- $\widehat{\delta}(q_0, 1100) = \delta(q_2, 0) = q_1$

となる. こうして, 1100 は M によって受理されることがわかった.

定義 7.4. 決定性有限オートマトン $M = (K, \Sigma, \delta, q_0, F)$ が文字列 $w = w_1 w_2 \cdots w_n \in \Sigma^*$ $(w_i \in \Sigma)$ を **受理する** とは, 拡張された遷移関数 $\widehat{\delta}$ によって, $\widehat{\delta}(q_0, w) \in F$ となるときをいう. M が w にを受理しないとき, M は w を **拒否する** という. M によって受理される文字列全体を $\mathcal{T}(M)$ と表し, これを **M が認識する言語** という.

例 7-5　決定性有限オートマトン $M = (K, \Sigma, \delta, q_0, F)$ が, $K = \{q_0, q_1\}$, $\Sigma = \{0, 1\}$, $F = \{q_0\}$ として, $\delta : K \times \Sigma \to K$ を表のように与える.

δ	0	1
$\to {}^*q_0$	q_0	q_1
q_1	q_0	q_1



このとき, M の開始状態は受理状態でもあるので, 空文 ε を受理する. また, M は 0 で終わる文字列を受理し, 1 で終わる文字列を入力すると状態は q_1 となってしまうから, $\mathcal{T}(M)$ は次のようになる.

$$\mathcal{T}(M) = \{w \in \Sigma^* \mid w = \varepsilon \text{ であるか, } w \text{ は 0 で終わる } \}$$

• 7-3 : 決定性有限オートマトンの設計

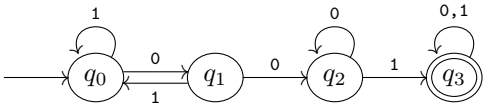
決定性有限オートマトン M があったとき, それが認識する言語について述べてきたが, 逆に「このような文字列を受理したい」といったような言語を認識する決定性有限オートマトンを設計することはできるだろうか. 例えば, 次のような問題を考えよう.

> $\Sigma := \{0, 1\}$ として, Σ 上の言語
>
> $$\mathcal{L} = \{w \in \Sigma^* \mid w \text{ は 001 を含む } \}$$
>
> について, $\mathcal{T}(M) = \mathcal{L}$ となるような決定性有限オートマトン M を設計せよ.

例えば, 0010 や 11000100 などは受理するが, 1101 や 0000 などは受理しないようなものを設計したい. 文字列 w に 001 が含まれるかを認識するまでに, 次のような段階を踏むことになる.

- 1 文字も読み込まない.
- 0 を認識する.
- 00 を認識する.
- 001 を認識する.

そこで, これらを 4 つの状態 q_0, q_1, q_2, q_3 に対応させる. 例えば, 初期状態を q_0 として, その後 0 を認識すれば状態 q_1 に遷移し, 1 を認識すれば状態 q_0 に遷移する. 状態 q_1 のときに 0 を認識すれば状態 q_2 に遷移, 1 を認識すれば状態 q_0 に遷移する. また, 状態 q_1 で 0 を認識すれば状態 q_2 に遷移し, 次に 1 を認識すれば状態 q_3 に遷移, 0 を認識すれば, 00 と続いているので状態 q_2 に遷移する. これらの状態遷移図をかけば次のようになるだろう.



すなわち, 次の表で得られる決定性有限オートマトンが 001 を含む文字列全体からなる言語を認識する決定性有限オートマトンである.

δ	0	1
$\to q_0$	q_1	q_0
q_1	q_2	q_0
q_2	q_2	q_3
${}^*q_3$	q_3	q_3

小テスト 7-1　決定性有限オートマトン M を ① で与えたものとする.

(1) $\widehat{\delta}(q_0, \texttt{0011010011})$ の値を計算せよ.
(2) M が認識する言語 $\mathcal{T}(M)$ を求めよ.

小テスト 7-2　次の決定性有限オートマトンが受理する文字列をすべて選べ. 集合 K, Σ, F を

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

として, 遷移関数：$\delta : K \times \Sigma \to K$ を以下のように与える.

δ	0	1
$\to q_0$	q_1	q_3
q_1	q_1	q_2
${}^*q_2$	q_1	q_2
q_3	q_3	q_3

(a) 00　　(b) 01　　(c) 11110000
(d) 0101001　　(e) 0101010101　　(f) 01010110001

演習問題 7-1　$\Sigma = \{\texttt{0}, \texttt{1}\}$ であるような決定性オートマトン M であって,

$$\mathcal{T}(M) = \{w \in \Sigma^* \mid \ell(w) > 0,\ w \text{ には } \texttt{0} \text{ が } 1 \text{ 個以上で } 3 \text{ の倍数個ある } \}$$

となるものを設計し, それの状態遷移図をかけ.

演習問題 7-2　$\Sigma = \{\texttt{a}, \texttt{b}\}$ であるような決定性オートマトン M であって,

$$\mathcal{T}(M) = \{w \in \Sigma^* \mid \ell(w) > 0,\ w \text{ には, まず } \texttt{a} \text{ が } 1 \text{ 個以上並び, その後 } \texttt{b} \text{ が } 1 \text{ 個以上並ぶ } \}$$

となるものを設計し, それの状態遷移図をかけ.

演習問題 7-3　次の決定性有限オートマトンが認識する言語を求めよ.

δ	0	1
$\to q_0$	q_2	q_0
${}^*q_1$	q_1	q_1
q_2	q_2	q_1

- **7-A：文字列全体のなす代数系**

Σ をアルファベットとする．Σ 上の文字列全体の集合 Σ^* 上の二項演算 $\cdot$ を次のように定義しよう．$w, w' \in \Sigma^*$ に対して

$$w \cdot w' = ww.$$

例えば，$w = \mathtt{01}$, $w' = \mathtt{01100}$ ならば，$ww' = \mathtt{0101100}$ といった具合に文字列をつなげる．このような $\cdot$ を **連接** と呼ぶ．

代数系 $(\Sigma^*, \cdot)$ は，文字列のつなげ方に順序は関係ないので結合法則を満たす．つまり，任意の $w, w', w'' \in \Sigma^*$ に対して

$$(w \cdot w') \cdot = w \cdot (w' \cdot w'')$$

を満たす．

また，空語 ε と任意の文字列 $w \in \Sigma^*$ との連接は

$$\varepsilon \cdot w = w \cdot \varepsilon = w$$

となるため，ε は代数系 $(\Sigma^*, \cdot)$ の単位元となる．

このように，代数系 $(G, *)$ が結合法則と単位元 e をもつとき，$(G, *)$ は **モノイド** と呼ばれる．上で述べたことから，次の命題が従うことになる．

命題. Σ をアルファベットとする．Σ^* 上の二項演算 $\cdot$ を文字列の連接と定めると，代数系 $(\Sigma^*, \cdot)$ はモノイドである．

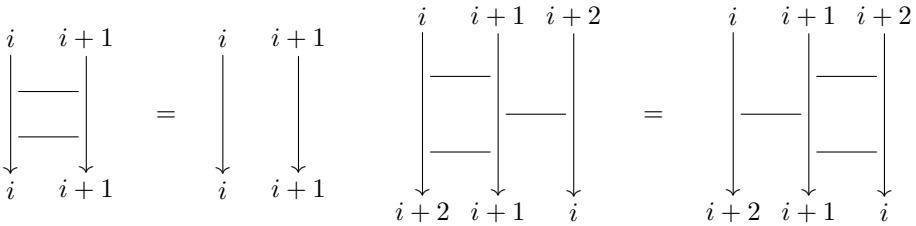
このような，あるアルファベット上の文字列全体に連接という二項演算を考えたものは計算機科学や数学では非常に重要な概念となっている．例えば，数学における「ホモロジー」と呼ばれる概念は「図形の穴の状態」を数学的に定式化したものであるが，これは代数系 $(\Sigma^*, \cdot)$ によって導入される概念である．更に，任意の群は，あるアルファベット Σ による代数系 $(\Sigma^*, \cdot)$ にある同値関係を定義した商集合と同じになる．例えば対称群 S_n は $\Sigma := \{s_1, \ldots . s_{n-1}\}$ として，同値関係を

$$s_i s_i \sim \varepsilon, \quad s_i s_{i+1} s_i \sim s_{i+1} s_i s_{i+1}, \quad s_i s_j \sim s_j s_i \ \ (|i-j| \geq 2)$$

によって定められるものを考え，これによる商集合 $\Sigma^*/\sim$ として実現される．この同値関係は天から降ってきたものではなく，

s_i：i 番目の縦棒と $i+1$ 番目の縦棒をつなぐ横線

に対応していて，はじめの 2 つの関係をあみだくじで表せば以下のようになる．



また，3 つめの関係は，i と j が 2 以上離れているとき，以下の等号が成り立つ．