ま　え　が　き

NHK の番組で “オートマタの世界” という古いオルゴールを紹介する番組があった。オートマタ (automata) はオートマトン (automaton) の複数形である。一般視聴者向けの番組名に私の専門に関わる用語がでてきたので非常に興味深く見た記憶がある。今では “オートマトン” と辞書を引いても “コンピュータの数学的なモデル” と説明していることも多いが，もともとのオートマトンの意味は “自動機械” である。オルゴールは音楽を奏でる自動機械 (オートマトン) であり，コンピュータは計算をする自動機械 (オートマトン) である。

コンピュータの数学的なモデルとしてのオートマトンに関する研究は，コンピュータをつくる気運が高まった 1930 年代頃に，“計算をするとはどういうことか” という本質的な問いから始まった。特に，数学者であるチューリング (Alan Mathison Turing) によって提案されたチューリング機械は，現在のコンピュータのモデルそのものである。そのほかに，線形拘束オートマトン，プッシュダウンオートマトン，有限オートマトン等があり，これら 4 種のオートマトンの能力の違い等を研究するのがオートマトン理論といわれる学問分野である。

一方 1950 年代の中頃，言語学者であるチョムスキー (Avram Noam Chomsky) によって，言語を生成するシステムとしての形式文法が定式化され，句構造文法，文脈依存文法，文脈自由文法，正規文法の 4 種の文法が定義された。形式文法によって生成される言語は形式言語と呼ばれるが，形式言語や 4 種の形式文法の生成能力の違い等を研究するのが形式言語理論といわれる学問分野である。

オートマトン理論には理科系のイメージがあり，形式言語理論には文科系のイメージがある。それぞれ研究が始まった時代も違えば，研究者の目的や立場も大きく異なる。どちらもまったく無関係のように見えるが，不思議なことに，両者は密接な関係にあったのである。

オートマトン理論と形式言語理論は，両者の密接な関係からオートマトン・形式言語理論とまとめて呼ばれることもある。オートマトン・形式言語理論は情報工学，情報科学の理論の中でも特に基礎となる理論であり，その考え方や用語は情報工学，情報科学の多くの分野で当然のように用いられている。

本書は大学，短大，高専等におけるオートマトン・形式言語理論に関する講義に使っていただけるテキストとして著したつもりである。

オートマトン・形式言語理論が扱う題材は多岐にわたるが，本書では題材を絞り，その分わかりやすく説明することとした。それだけでも，情報工学あるいは情報科学の基礎としてのオートマトン・形式言語理論の考え方が理解でき，その美しさと面白さを十分に味わっていただけるものと考えている。本書が扱うのは理論であるが，他の理論に関する本のように証明は一切でてこない。できるだけ多くの例を用い，わかりやすい説明を加え，直観的に考え方を理解していただけるように心がけたつもりである。

ICT (Information and Communication Technology) 社会といわれる現代においては，大学，短大，高専等で学ぶ学生・生徒諸君は，理科系であれ文科系であれ，コンピュータと無関係でいることはできない。大多数の人が今後ずっとコンピュータと関わっていくことになる。学生・生徒諸君の 1 人でも多くの人に，オートマトン・形式言語理論の美しさと面白さを理解していただいて，ますます進化する ICT 社会で生きていくための素養を身に付けていただきたい。本書が少しでもそのお役に立てば幸いである。

最後になるが，私をこのオートマトン・形式言語理論という楽しい学問の道に導いて下さった恩師である本多波雄先生，ならびに米田政明先生に心から感謝申し上げる。

また，本書の出版にご尽力下さった大同大学教授 佐藤秀樹先生，株式会社コロナ社に感謝申し上げる。

2014 年 2 月

広 瀬 貞 樹

目　　　次

1.　形　式　言　語

2.　有限オートマトン

3.　プッシュダウンオートマトン

4.　チューリング機械と線形拘束オートマトン

5.　形　式　文　法

6.　有限オートマトンと正規文法

7.　プッシュダウンオートマトンと文脈自由文法

8.　線形拘束オートマトンと文脈依存文法:
チューリング機械と句構造文法

1 形　式　言　語

言語とは何かと問われると，日本語とか，英語とか，フランス語のような言語を連想する人が多いと思われる。このような人間が生活を送る中で自然発生的に生まれ使用されている言語を**自然言語** (natural language) と呼ぶ。また，情報やコンピュータに関係する人ならば，C 言語，Java 等のプログラミング言語を連想する人も多いと思われる。このような人間が何かの目的で使用するために定義した言語を**人工言語** (artificial language) と呼ぶ。

これから学ぶ**形式言語** (formal language) は，自然言語，人工言語を抽象化した言語，またはこれらの数学的なモデルというようなものであり，それゆえ**数理言語** (mathematical language) と呼ばれることもある。

本章では，本書の主対象である形式言語がどのようなものであるのか，また，形式言語の識別機械であるオートマトンや形式言語の生成機械である形式文法がどのようなものなのか説明する。

1.1　形　式　言　語

記号の任意の有限集合†を**アルファベット** (alphabet) と呼ぶ。形式言語を考えるうえで使ってよい記号の集合である。アルファベットは一般的にギリシャ文字の大文字 Σ (シグマ) で表すことが多い。例えば，英語であれば $\Sigma = \{a, b, c, \cdots, z\}$ であろうし，日本語 (ひらがな) であれば $\Sigma = \{$ あ, い, う, $\cdots$, ん $\}$ であろう。また，2 進数を考えるのであれば $\Sigma = \{0, 1\}$ であろう。

† 集合については**付録 A.1** を参照されたい。

アルファベットを Σ として，Σ に属する記号を (重複を許して) 並べた任意の記号列を，Σ の上の**語** (word) と呼ぶ。例えば $\Sigma = \{a, b\}$ としたとき，$a, ba, aaba, bbaabab$ 等は Σ の上の語の例である。記号を 0 個並べた記号列も 1 つの特別な語と考えて，これを**空語** (empty word) と呼ぶ。本書では空語を ε (イプシロン) で表す。なお，語に含まれている記号の数をその**語の長さ** (length of the word) といい，語 w の長さを $|\, w \,|$ で表す。例えば，$|\, aaba \,|= 4$, $|\, bbaabab \,|= 7$ であり，$|\, \varepsilon \,|= 0$ である。

Σ の上の語の任意の集合を Σ の上の**言語** (language) という。アルファベット Σ が明らかでいちいち断る必要がないときは，Σ の上の語や Σ の上の言語を，単に語や言語といってもよい。

Σ の上のすべての語の集合も Σ の上の 1 つの言語であり，これを Σ^* で表す (シグマ・スターまたはシグマ・アスタリスクと読む)。例えば，$\Sigma = \{a, b\}$ とすると $\Sigma^* = \{\varepsilon, a, b, aa, ab, ba, bb, aaa, \cdots\}$ である。また，空語 ε 以外のすべての語の集合もよく使われる 1 つの言語であり，これを Σ^+ で表す (シグマ・プラスまたはシグマ・ダガーと読む)。すなわち，$\Sigma^+ = \Sigma^* - \{\varepsilon\}$ であり，例えば，$\Sigma = \{a, b\}$ とすると $\Sigma^+ = \{a, b, aa, ab, ba, bb, aaa, \cdots\}$ である。さらに，語を 1 つも含まない空集合 $\{\} = \emptyset$ も 1 つの言語である。空語 ε は立派な 1 つの語であるから，$\{\}$ と $\{\varepsilon\}$ は異なることに注意されたい。

Σ の上の語の任意の集合が Σ の上の言語であると述べたが，言い方を変えると，Σ^* の任意の部分集合が Σ の上の言語であるといえる (**図 1.1**)。

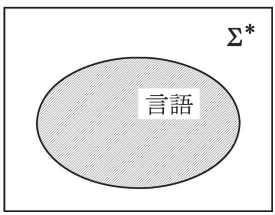

図 1.1　Σ の上の言語

例題 1.1　言語の例をいくつか見てみよう。それぞれがどのような言語であるか各自確かめてみよう。解答は，次ページに示す。

(1) アルファベットが 1 記号のみからなる最も単純な場合を考えよう。$\Sigma = \{a\}$ とする。

$$L_{1.1} = \{\varepsilon, a, aa, aaa, aaaa, aaaaa, aaaaaa\}$$

$$L_{1.2} = \{\varepsilon, aa, aaaa, aaaaaa, aaaaaaaa, \cdots\}$$

$$L_{1.3} = \{a, aaa, aaaaa, aaaaaaa, aaaaaaaaa, \cdots\}$$

$$L_{1.4} = \{\varepsilon, aa, aaa, aaaa, aaaaaa, aaaaaaaa, aaaaaaaaa, \cdots\}$$

$$L_{1.5} = \{aa, aaa, aaaaa, aaaaaaa, aaaaaaaaaaa, \cdots\}$$

$$L_{1.6} = \{\varepsilon, a, aaaa, aaaaaaaaa, aaaaaaaaaaaaaaaa, \cdots\}$$

$$L_{1.7} = \{a, aa, aaaa, aaaaaaaa, aaaaaaaaaaaaaaaa, \cdots\}$$

等は，$\Sigma = \{a\}$ の上の言語である。

(2) アルファベットが 2 記号からなる場合を考えよう。$\Sigma = \{a, b\}$ とする。

$$L_{1.8} = \{a, aa, ba, aaa, aba, baa, bba, \cdots\}$$

$$L_{1.9} = \{bb, abb, bab, bba, aabb, abab, abba, baab, baba, bbaa, \cdots\}$$

$$L_{1.10} = \{ab, aabb, aaabbb, aaaabbbb, \cdots\}$$

$$L_{1.11} = \{aba, abaa, abaaa, \cdots, aabba, aabbaa, aabbaaa, \cdots, aaba, aaaba, \cdots, abbaa, aaabbaa, aaaabbaa, \cdots\}$$

$$L_{1.12} = \{ab, ba, aabb, abab, abba, baab, baba, bbaa, \cdots\}$$

$$L_{1.13} = \{a, b, aa, bb, aaa, aba, bab, bbb, \cdots\}$$

$$L_{1.14} = \{aa, bb, aaaa, abba, baab, bbbb, \cdots\}$$

$$L_{1.15} = \{aa, bb, aaaa, abab, baba, bbbb, \cdots\}$$

等は，$\Sigma = \{a, b\}$ の上の言語である。

【解答】 例題 1.1 に示した言語はそれぞれ以下のような言語である。

(1) アルファベットが 1 記号のみからなるので，言語の要素である語の長さの違いということになる。同じ記号 a を n 個並べた記号列を a^n と書く。すなわち，$a^0 = \varepsilon, a^1 = a, a^2 = aa, a^3 = aaa, \cdots$ である。

$$L_{1.1} = \{a^0, a^1, a^2, a^3, a^4, a^5, a^6\} = \{a^n \mid 6 \geq n\}$$
$$L_{1.2} = \{a^0, a^2, a^4, a^6, a^8, \cdots\} = \{a^n \mid n \text{ は偶数 }\}$$
$$L_{1.3} = \{a^1, a^3, a^5, a^7, a^9, \cdots\} = \{a^n \mid n \text{ は奇数 }\}$$
$$\begin{aligned} L_{1.4} &= \{a^0, a^2, a^3, a^4, a^6, a^8, a^9, \cdots\} \\ &= \{a^n \mid n \text{ は 2 の倍数または 3 の倍数 }\} \end{aligned}$$
$$L_{1.5} = \{a^2, a^3, a^5, a^7, a^{11}, \cdots\} = \{a^n \mid n \text{ は素数 }\}$$
$$L_{1.6} = \{a^0, a^1, a^4, a^9, a^{16}, \cdots\} = \{a^n \mid n \text{ は平方数 }\}$$ [†1]
$$L_{1.7} = \{a^1, a^2, a^4, a^8, a^{16}, \cdots\} = \{a^{2^n} \mid n \geq 0\}$$

(2) アルファベットが 2 記号になっただけで，かなり複雑な言語が構成できる。いずれもオートマトン・形式言語理論に関するどの本にもほとんど出てくるような代表的な言語である。

$$L_{1.8} = \{w \mid w \in \Sigma^+,\ w \text{ の末尾が } a \text{ である }\}$$
$$L_{1.9} = \{w \mid w \in \Sigma^+,\ w \text{ はちょうど 2 個の } b \text{ を含む }\}$$
$$L_{1.10} = \{a^n b^n \mid n \geq 1\}$$
$$\begin{aligned} L_{1.11} = \{a^n b^m a^l \mid\ & n \geq 1, m \geq 1, l \geq 1 \text{ であり，} \\ & \text{かつ } n = m \text{ または } m = l \text{ である }\} \end{aligned}$$
$$L_{1.12} = \{w \mid w \in \Sigma^+,\ w \text{ に含まれる } a \text{ の数と } b \text{ の数が同じ }\}$$
$$L_{1.13} = \{w \mid w \in \Sigma^+,\ w = w^R\}$$ [†2]
$$L_{1.14} = \{ww^R \mid w \in \Sigma^+\}$$
$$L_{1.15} = \{ww \mid w \in \Sigma^+\}$$

◇

†1 **平方数** (square number) とは，ある整数の 2 乗 (平方) で表される整数のことである。

†2 w^R は w を**逆順** (<u>R</u>everse) にした記号列を表す。すなわち，$w = a_1 a_2 \cdots a_n$ とするとき，$w^R = a_n \cdots a_2 a_1$ である。$w = w^R$ が成り立つということは，w を左から読んでも右から読んでも同じであるということである。$L_{1.13}$ は，いわゆる**回文** (palindrome) からなる言語である。

1.2　形式言語の識別機械と生成機械

言語を理解している，言語がわかっているというのは，いったいどういうことであろうか。

例えば，子供に日本語の文章を見せ，これは正しい日本語であるかどうか聞いてみる。正しい日本語の文章を見せたときはいつも“正しい”と答え，正しくない日本語の文章を見せたときはいつも“正しくない”と答える。何度聞いても一度も間違えることなく答えられる。そのうち，この様子を見ていた大人たちは，この子供は日本語を理解している，日本語がわかっていると思うのではないだろうか。この子供は，正しい日本語であるのかそうでないのかを識別できると考えられたからである (**図 1.2**†)。

図 1.2　正しい日本語を識別できる子供

また例えば，子供が日本語をしゃべり続けているとする。ずっとしゃべり続けている。しかし，一度も間違えない。変な日本語をしゃべることはまったくない。すらすらと流暢（りゅうちょう）に正しい日本語をしゃべり続ける。しばらくすると，この子供がしゃべるのを聞いていた大人たちは，この子供は日本語を完全に理解している，日本語がわかっていると思うのではないだろうか。この子供は，正しい日本語をつくり出すことができる，生成することができると考えられたからである (**図 1.3**)。

† イラストはイラストレータ カシワギマリさんによる。

図 1.3　正しい日本語を生成できる子供

先にも述べたが，本書の主対象は 1.1 節で説明した形式言語である。

オートマトンは形式言語に対する上述した前の子供のようである。ある形式言語について，その要素である語をオートマトンに与えると “正しい” と答え，その要素でない語を与えると “正しくない” と答えることができるのである。そうであれば，オートマトンはその形式言語を理解している，形式言語がわかっていると考えてよい。すなわち，オートマトンは形式言語を識別する**識別機械** (recognizer)，あるいは**識別システム**である。

一方，形式文法は形式言語に対する上述した後の子供のようである。ある形式言語について，形式文法がその要素であるすべての語を生成することができるのである。そうであれば，形式文法はその形式言語を理解している，形式言語がわかっていると考えてよい。すなわち，形式文法は形式言語を生成する**生成機械** (generator)，あるいは**生成システム**である。

識別機械としてのオートマトンには，4 つのモデル†がある。コンピュータを作る機運が高まった 1930 年代の中頃に，コンピュータの数学的なモデルとして**チューリング** (Alan Mathison Turing) によって提案されたチューリング機械のほかに，線形拘束オートマトン，プッシュダウンオートマトン，有限オートマトンがある。これら 4 種のオートマトンの言語の識別能力の違い等を研究するのが**オートマトン理論** (automata theory) といわれる学問分野である。

† 基本的に 4 つという意味で，ほかにもいろいろな変種のモデルが提案されており，それらの言語の識別能力についても研究されている。

一方，生成機械としての形式文法にも，4 つの型がある。1950 年代の中頃，言語学者である**チョムスキー** (Avram Noam Chomsky) は自然言語 (英語) の研究に関連し，生成機械としての形式文法を定式化した。彼は，形式文法にいくつかの制限を加えて，句構造文法，文脈依存文法，文脈自由文法，正規文法の 4 つの型の形式文法を定義した。これら 4 種の形式文法の言語の生成能力の違い等を研究するのが**形式言語理論** (formal language theory) といわれる学問分野である。

オートマトンには理科系のイメージがあり，形式文法には文科系のイメージがある。また，上でも述べたように，オートマトン理論に関する研究は 1930 年代中頃から始まり，形式言語理論に関する研究は 1950 年代中頃から始まっている。時代も違えば，研究者の目的や立場も大きく異なる。どちらもまったく無関係のように見えるが，不思議なことに，両者は密接な関係にあったのである。

以下に示すように，一番簡単な識別機械である有限オートマトンの識別能力と，形式文法に一番強い制限が加わった正規文法の生成能力がまったく等しい。また，次に簡単な識別機械であるプッシュダウンオートマトンの識別能力と，形式文法に次に強い制限が加わった文脈自由文法の生成能力がまったく等しい。さらに，線形拘束オートマトンの識別能力と文脈依存文法の生成能力が等しく，チューリング機械の識別能力と句構造文法の生成能力も等しいのである。

$$\begin{array}{rcl} \text{有限オートマトン} & \equiv & \text{正規文法} \\ \text{プッシュダウンオートマトン} & \equiv & \text{文脈自由文法} \\ \text{線形拘束オートマトン} & \equiv & \text{文脈依存文法} \\ \text{チューリング機械} & \equiv & \text{句構造文法} \end{array}$$

それではこれから 2 章以降で，有限オートマトン，プッシュダウンオートマトン，線形拘束オートマトン，チューリング機械や正規文法，文脈自由文法，文脈依存文法，句構造文法がどのようものか，またそれらの関係について，具体的に見ていくことにしよう。

索　　引

—— 著 者 略 歴 ——

1980 年　東北大学大学院工学研究科博士課程修了（情報工学専攻）
　　　　工学博士
1980 年　株式会社富士通研究所勤務
1984 年　神奈川大学工学部助教授
1989 年　富山大学工学部助教授
1998 年　富山大学工学部教授
2008 年　富山大学工学部長
2011 年　富山大学理事・副学長
　　　　現在に至る

著　書：情報処理の基礎（共著），共立出版（1989）
　　　　アルゴリズムの基礎（共著），共立出版（1997）
　　　　オートマトン・言語理論の基礎（共著），近代科学社（2003）
　　　　あるごりずむ（単著），近代科学社（2006）
　　　　オートマトン・言語理論入門（共著），共立出版（2012）

著者のキャラクター（イラスト）は富山大学工学部知能情報工学科の
卒業生による。

オートマトン・形式言語理論
Automata and Formal Language Theory

2014 年 4 月 7 日　初版第 1 刷発行　★

検印省略

著　　者　広　瀬　貞　樹
発 行 者　株式会社　コ ロ ナ 社
　　　　　代 表 者　牛 来 真 也
印 刷 所　三 美 印 刷 株 式 会 社

112–0011　東京都文京区千石 4–46–10
発行所　株式会社　コ　ロ　ナ　社
CORONA PUBLISHING CO., LTD.
Tokyo Japan
振替 00140–8–14844・電話（03）3941–3131（代）
ホームページ http://www.coronasha.co.jp

ISBN 978－4－339－02476－0（安達）　（製本：愛千製本所）
Printed in Japan