数理解析研究所講究録
第1915巻 2014年 100-112

双対性による正規言語のVariety Theory

浦本 武雄

京都大学・数学教室

概要

本稿は、理論計算機科学における一分野である「正規言語のvariety theory」を、他分野の数学者に向けて概説する目的で書かれている。正規言語とは、その所属問題が有限オートマトンで解けるような言語(文字列の集合)を言い、正規表現と呼ばれる簡明な記法を持つことと併せて、文字列の置換やパターン検索といったプログラミングに係るテキスト処理技術に広く利用される。一方、正規言語のvariety theoryとは、正規言語、有限オートマトン、および有限モノイドの間の密接な関係性を体系化した理論として1975年にEilenbergによって創出されたものであり、以来今日まで、代数学的・幾何学的・論理学的方面からの幅広い研究がされて来た。近年、正規言語のvariety theoryは、Stone双対性の観点から再証明・再構築されつつあり、本稿もその文脈に位置づけられる。

1 はじめに

一般に**言語**(language)とは、有限長の語(word)の任意の集合を言い、中でも**正規言語**(regular language)とは有限オートマトンによって受理できる言語のことを言う(§2)。正規言語の組合せ的性質の研究は、理論的にも興味深いものであるのみならず、**正規表現**(regular expression)を使ったテキスト処理技術[6]とも関連があり、その興味は決して理論的なものに留まらない。

特に、本稿のテーマでもある**正規言語のvariety theory**は、以下三者の間に観られる密接な関係性を使うことで、正規言語の組合せ的性質の決定可能性(decidability)を証明する為の重要なアプローチを与える:

1. 正規言語の組合せ的性質
2. 有限モノイドの代数的性質
3. 有限オートマトンの幾何的性質

正規言語のvariety theoryの直接の起源は、1965年のSchützenbergerの研究にさかのぼり、以来、上記三者の関係性を示唆する多くの具体的類似結果が示された[2, 14]。この三者の対応関係を、具体例でではなくより公理的に体系化したのが、代数的位相幾何学でも有名なS. Eilenbergであり、彼の著書[4, 5]にまとめられている。

近年では、正規言語のvariety theoryを双対定理の観点から、より洗練された形で再証明・再構築する研究が進んでおり、本稿は特にその文脈に属する。variety theory再構築の発端となったのは1997年のPippengerによる研究[10]で、それは後にGehrkeら[7]およびRhodesら[12]によって引き継がれた。彼らの研究によると、上記のうち特に上二つ:

1. 正規言語の組合せ的性質
2. 有限モノイドの代数的性質

の間の対応関係は、(1′) 正規言語の成す**双代数**と(2′) **副有限モノイド**の間のStone双対定理[1]の帰結として再証明できる(§3)。

本稿では、残る一つのもの(3. 有限オートマトンの幾何的性質)を込めた全三者の対応関係を、適切

[1]ブール代数の成す圏とStone空間の成す圏の間の反辺同値

な双対定理の帰結として再証明し、Gehrke らによる研究に欠けている部分を埋める。結論から言えば、ブール代数の双代数と副有限モノイドに加え、必要なものは副有限モノイドの表現の圏の特徴付けにある。より具体的に言うと、その特徴付けとは、副有限群の表現の圏である**ガロア圏**(Galois category) の公理を弱めたものであり、本稿はその公理を満たす圏が常に副有限モノイドの表現の圏と同値となることを示している。

勿論、こういった結果はあくまでも、正規言語の variety theory を深めるものというよりはむしろ、見通しを良くするためのものであることに注意しなければならない。また、見通しを良くした先に望まれるのは、それまでは関連が明らかではなかった既存の知識同士を有機的につなげ、研究の交流を生むことにある。本稿が variety theory の専門家向けの論文というよりはむしろ、より広く一般向けの解説記事として書かれているのは、その点を考慮したからに他ならない。

本稿のテーマである正規言語の variety theory は、しかしながら、本稿の分量で解説しきれるほど浅くはない。また、本稿では証明には立ち入らず、必要な箇所で参考文献を挙げるに留めている。本稿は、正規言語と有限オートマトンに関する基本的・具体的な話題から始め、徐々に正規言語の構造を巡る抽象性の高い数学に踏み入っていく様に書かれている。

2 正規言語と有限オートマトン

例えば「与えられた二進数 $b_0b_1\cdots b_N$ が素数であるか」という問題には、勿論、それを判定するための**アルゴリズム**が存在する。つまり、決められた手順 (アルゴリズム) に従って処理すれば、どんな二進数 $b_0b_1\cdots b_N$ でも素数であるか否かが機械的に分かるということだ。

このように**語**(有限長の文字列) を入力としてその性質を判定する問題を、**語の所属問題**(membership problem) と言い、その問題を判定するアルゴリズムが存在するとき、その問題は**決定可能**(decidable) であるという。より形式的には、語の所属問題とは「与えられた語 w が、(何らかの性質を満たす) 語から成る特定の集合 L に属すか否か」を判定する問題のことと言っても良い[2]。

語の集合 L を一つ固定したとき「与えられた語 w が L に属すか否か」を判定する問題は、L に関する最も基本的な問題の一つであり、L の構造によって L への所属問題を決定する困難さが異なる。**正規言語**とは特に、有限オートマトンと呼ばれる一種の有向グラフを使ってその所属問題が決定できる様な語の集合 (**言語**) を言い、本稿の主要な関心の対象である。

2.1 有限オートマトンと受理言語

以下では A を固定された有限集合とし、**アルファベット**(alphabet) と呼ぶ。また、A 上の**語**(word) とは、A の元を有限個並べた任意の列 $a_1a_2\cdots a_n$ を言い、A^* と書いて A 上の語全体の集合を表すことにする。特に $\varepsilon \in A^*$ を長さ 0 の語とし、**空語**(empty word) と呼ぶ。

定義 1. A 上の**言語**(language) とは、A^* の任意の部分集合 $L \subseteq A^*$ のことを言う。

例えば：

$$L_1 := \Big\{ w \in \{0,1\}^* \mid \|w\|_1 \equiv 1 \ (\mathrm{mod}3) \Big\}.$$

はバイナリ・アルファベット $A = \{0,1\}$ 上の言語の例となる。ここで、一般にアルファベット A 上の語 w と文字 $a \in A$ に対し $\|w\|_a$ と書いて、w の中に現れる文字 a の個数を表す。この例では、例えば 0010111 や 0100 は L_1 に属し、1100 や 10011 は L_1 に属さない。

また、この言語 L_1 に対する語の所属問題は勿論、決定可能であることも分かる。実際、0 と 1 から成る語 w が与えられた時、その中に現れる文字 '1' の個数を数え、それが 3 を法として 1 と合同であるか

[2]例えば二進数の素数判定問題は勿論、「与えられた二進数 $b_0b_1\cdots b_N$ が、素数である二進数の全体から成る集合に属すか」と単純に言い換えられる。

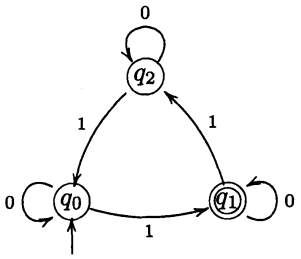
を判定するアルゴリズムを書くことは決して難しくない。しかしもう少し踏み込んで言えば、この言語への所属問題は、その判定に高級なアルゴリズムを使う必要はなく、特に**有限オートマトン**と呼ばれる一種の有向グラフを使った簡単な方法で決定可能な例になっている。

定義 2. A 上の**有限オートマトン**(finite automaton)とは、以下のものの4つ組 $\mathfrak{A} = (Q, \delta, q_0, F)$ を言う：

- **状態**(state) の有限集合 Q；
- **遷移関数**(transition function) $\delta : Q \times A \to Q$；
- **初期状態**(initial state) $q_0 \in Q$；
- **終状態**(final state) の集合 $F \subseteq Q$.

以下では、$\delta(q, a) =: q \cdot a$ と書く。また、語 $w \in A^*$ に対して、$q \cdot \varepsilon := q$ および $q \cdot (wa) := (q \cdot w) \cdot a$ によって帰納的に $q \cdot w \in Q$ を定める。

通常、有限オートマトンは、例えば以下のような有向グラフの形式で表記することが多い。



(1)

つまり、有限オートマトン (Q, δ, q_0, F) が与えられた時、Q の元を頂点 (vertex) とし、二つの頂点 $q, q' \in Q$ に対し辺 $q \xrightarrow{a} q'$ があるのは、$q' = q \cdot a$ の時とする。各辺は A の元で彩色されていて、各頂点 $q \in Q$ と各文字 $a \in A$ に対して、$q \xrightarrow{a} q'$ なる頂点 q' がちょうど一つ存在する ($q' = q \cdot a$)。その有向グラフの中で、とくに初期状態 $q_0 \in Q$ には始点のない矢印 ($\to\bigcirc$) をつけ、また終状態 $q \in F$ を二重丸にすることで他の状態と区別している。

一般に A 上のオートマトン (Q, δ, q_0, F) と語 $w = a_1 a_2 \cdots a_n \in A^*$ に対して、状態 q から順に文字 $a_1, a_2, \cdots, a_n$ で彩色された辺をたどって行って状態 q' にたどり着くとき、$q \xrightarrow{w} q'$ と書く (勿論これは、定義から $q \cdot w = q'$ の時に他ならない)。例えば上のオートマトンの例で言うと：

$$q_0 \xrightarrow{1101} q_0$$

であるし、また：

$$q_0 \xrightarrow{101} q_2$$

であることも見て取れる。さらに言うと、このオートマトンでは、初期状態 q_0 から出発して終状態 q_1 にたどり着くのは、語 $w \in \{0,1\}^*$ がちょうど L_1 の元であるときに他ならない。つまり：

$$L_1 = \{w \in \{0,1\}^* \mid q_0 \xrightarrow{w} q_1\}. \tag{2}$$

となる。

単純に見えるこの事実の重要な点は、この事実から、言語 L_1 への語の所属問題を解くアルゴリズムが得られるという点にある。実際 (2) から、語の所属問題「$w = a_1 a_2 \cdots a_n \in L_1$ か否か」を判定するには、単純にオートマトン (1) の中で「q_0 から順に $a_1, a_2, \cdots, a_n$ で彩色された辺をたどって行って q_1 に到達するか否か」を確かめればよいと分かり、この手続きが「$w \in L_1$ か否か」を判定するアルゴリズムになる。

このように、言語 $L \subseteq A^*$ が、何らかの有限オートマトン (Q, δ, q_0, F) の初期状態から終状態へ至る経路上の語全体の言語と一致するとき、L への語の所属問題は決定可能であることが一般に従う。そのようなオートマトンが存在するような言語を、**正規言語**と言う。形式的には：

定義 3. A 上の言語 $L \subseteq A^*$ が**正規言語**(regular language) であるとは、A 上の有限オートマトン $\mathfrak{A} = (Q, \delta, q_0, F)$ が存在して：

$$L = \{w \in A^* \mid \exists q \in F.\ q_0 \xrightarrow{w} q\}. \tag{3}$$

となるときを言う。この時、L はオートマトン $\mathfrak{A}$ によって**受理**(accept) されるという。また、オートマトン $\mathfrak{A}$ が受理する言語 (上式の右辺) を、以下では $L(\mathfrak{A})$ と書く。

2.2 正規言語の形式的記法：正規表現

正規言語が実際に活用される場面は大小さまざまある。例えば、単純だが典型的な事例として、「あるフォルダ以下に格納されている pdf ファイルを列挙する」タスクを挙げることが出来る。実際、このタスクでは、対象フォルダ下に (もしかしたら大量に) あるファイルを一つ一つ読み込んで行き：

- ファイル名 (=英数字と諸記号からなる語) が ".pdf" という語で終わるか否か

を判定する問題を解く必要がある。これは勿論「.pdf という語で終わる」語の成す正規言語への所属問題に他ならない。他にも、サイズの大きいテキストファイルから特定の (或は複数の) キーワードを検出するタスクなど日常的なタスクもこの種の問題を含み、正規言語への所属問題に帰着されるタスクは決して少なくない。

正規言語に特別な関心が集まるのは、上記に加え、正規言語が**正規表現**という簡明な形式的記法を持つことにも依存している [6]。まず定義から言えば、正規表現とは以下で与えられる式のことを言う[3]。

定義 4. A をアルファベットとする。A 上の**正規表現**(regular expression) とは、以下の規則で帰納的に定義される式のことを言う：

1. 記号 $\emptyset$ および A の各元 $a \in A$ は正規表現；
2. p と q がともに正規表現のとき、$p+q$、pq、$p \wedge q$、p^c および p^* は正規表現.

例えば、A がバイナリ・アルファベット $\{0,1\}$ である時、形式的な式 $(01+10)^*(11+00)^*$ や $(0(10+11)^*)^c$ は A 上の正規表現になる。

さらに、正規表現 p が与えられたとき、帰納的に一つの言語 $R(p)$ を定義することができる。

定義 5. A 上の正規表現 p に対し、言語 $R(p) \subseteq A^*$ が以下の規則で定義される：

$$\begin{aligned} R(\emptyset) &:= \emptyset & R(a) &:= \{a\} \\ R(p+q) &:= R(p) \cup R(q) & R(pq) &:= R(p) \cdot R(q) \\ R(p \wedge q) &:= R(p) \cap R(q) & R(p^c) &:= A^* \backslash R(p) \end{aligned}$$

ここで、言語 $L, R \subseteq A^*$ に対し、$L \cdot R$ とは：

$$L \cdot R := \{uv \in A^* \mid u \in L \ \wedge \ v \in R\}$$

で定義される言語を表す。また $L^0 := \{\varepsilon\}$、$L^{n+1} := L^n \cdot L$ とするとき：

$$R(p^*) := \bigcup_{i=0}^{\infty} R(p)^n.$$

正規表現 p から得られる言語 $R(p)$ は、実は常に正規言語であって、逆に正規言語は必ずこの形で無くてはならないということが知られている [8]。

定理 1 (Kleene). 言語 $L \subseteq A^*$ が正規である為の必要十分条件は、A 上のある正規表現 p が存在して $L = R(p)$ となることである。

この事実を背景にして、正規言語を計算機上で実際に表現するときには、テキスト形式で表記しやすい正規表現を用いる [6]。

ここで、正規言語 L に対し $L = R(p)$ となる正規表現 p は、一般には複数ありうるという点に注意しなければならない。実際、例えばバイナリ・アルファベット上の空言語 $\emptyset \subseteq \{0,1\}^*$ には：

$$\begin{aligned} \emptyset &= R(\emptyset) \\ &= R\big((00+11)^* \wedge (10+01)^*\big) \end{aligned}$$

という二つの正規表現による表現がある。表す言語としては二つの正規表現は等価だが、重要なのは、実際に計算機で語の所属問題「$w \in R(\emptyset)$」と「$w \in R\big((00+11)^* \wedge (10+01)^*\big)$」を解くときの計算量が異なるという点にある。

これはちょうど、同じ関数でもそれを計算するプログラムには複数あって、しかも計算量に違いがあ

[3] 通常この式は「extended regular expression」と呼ばれるもので、普通の正規表現を拡張したものだが、後の都合上こちらを正規表現と呼ぶことにしている。普通の正規表現では、$p \wedge q$ と p^c という記法は使わない。(しかし表現できる言語は同じ。)

るのと似ている。そしてこのことは、正規言語への所属問題を効率よく解くには、筋の良い正規表現を使う必要があることを意味している。素朴に書き下したプログラムが(計算量などの観点から)ベストなものとは限らないのと同様、素朴に選んだ正規表現がベストなものとは限らないのだ。そのため、自然に問題となるのは「書き下した正規表現を(何らかの基準で)自動的に**最適化**できないか」ということになるが、**正規表現の variety theory** は、この種の問題に一定の方法論を与えてくれる。

3 正規言語の Variety Theory

正規言語の variety theory の直接の起源は、1965 年の Schützenberger の論文 [13] にさかのぼることが出来る。彼がその論文で示したのは、「正規言語が $*$ を使わない正規表現で書けること (**star-free**) と、その **syntactic monoid** (§3.1) が非自明な部分群を持たないこと (**aperiodic**) は同値である」という事実で、これは正規言語の構造に関するその後の研究に大きな影響を与えた。

「正規言語が、(何らかの) 良い性質 (star-free など) を満たす正規表現で書けるか」という問題は、正規表現の最適化の根本にある問題で一般には解くのが難しいが、この種の問題に対して、Schützenberger の結果は一つの指針を示しており、とても強力なアプローチとなっている。

3.1 正規言語の Syntactic Monoid

A を有限のアルファベットとし、A^* を A 上の語からなる集合とすると、A^* には自然にモノイド (monoid) の構造が入る[4]。

syntactic monoid とは、一般に言語 $L \subseteq A^*$ が与えられた時、以下で定義されるモノイドのことを言う。

定義 6. 言語 $L \subseteq A^*$ の **syntactic monoid** とは、モノイド A^* を以下の同値関係 $\equiv_L$ によって割った、商モノイド $A^*/\equiv_L$ を言う：

$$u \equiv_L v \quad \Leftrightarrow \quad \forall x, y \in A^*.\big(xuy \in L \Leftrightarrow xvy \in L\big).$$

以下では言語 L に対し、その syntactic monoid を $M(L)$ と書く。また、自然な射影を $\pi_L : A^* \twoheadrightarrow M(L)$ と書く。

言語の syntactic monoid の基本的な性質として、「その有限性が、元の言語の正規性が特徴づける」という点を挙げることが出来る [9]。

命題 1. 言語 $L \subseteq A^*$ が正規言語であることと、その syntactic monoid $M(L)$ が有限モノイドであることは同値である。

さらに、正規表現 p が与えられたとき、それが表す正規言語 $L = R(p)$ の syntactic monoid $M(L)$ の構造 (つまり、その積表) は計算可能であることも知られている。

この事実と Schuützenberger の定理を合わせると、重要な決定可能性を証明することが出来る。Schützenberger の定理は「正規言語が $*$ を使わない正規表現で書けることと、その syntactic monoid が非自明な部分群を持たないことは同値」というものであったが、それから「与えられた正規表現 p が、実は $*$ を使わない正規表現 q と等価 (同じ言語を定める) か」が決定可能であることが従う。実際、正規表現 p が与えられた時、その言語 $L = R(p)$ の syntactic monoid $M(L)$ の構造は計算可能であった。その積表を見れば $M(L)$ が非自明な部分群を含むか否かが分かるため、Schützenberger の定理から、結果的に $L = R(p)$ が $*$ を使わない正規表現 q で書けること (つまり $L = R(q)$)、すなわち p が $*$ を含まない q と等価であること (つまり $R(p) = R(q)$) も判定できると保証される。

3.2 正規言語の Variety

「与えられた正規言語 L が star-free であるか否か」を判定する問題は、一見して決定困難に見える。と

[4] つまり、二つの語 $u, v \in A^*$ の連接 uv を積とし、空語 ε が単位元となる。

いうのも、「L が star-free で**ない**」と結論づけるには、素朴には「(無限にある) $*$ を含まないどの正規表現でも書けない」ことを確かめる必要がありそうに見えるからだ。

しかし L の star-free 性を、その (有限の大きさの) syntactic monoid $M(L)$ の代数的性質「非自明な部分群を含まない」で特徴づけることで、正規言語の star-free 性の決定可能性を証明することが出来た。その他にも locally testability[2] や piecewise testability[14] といった、正規言語の組合せ的性質の決定可能性が、syntactic monoid を経由した同様の方法によって証明されている。

このように、正規言語の組合せ的性質は、場合によってはその syntactic monoid の代数的性質で特徴づけられることがある。これを利用して、正規言語の組合せ的性質の決定可能性を示すというのが、この方法の要になっている。こういったアプローチを公理的に体系化したのが Eilenberg であり、その際に**正規言語の variety** という概念が導入された。

定義 7. 正規言語の族 $\mathcal{V}$ が **variety** であるとは、各アルファベット A に対し、$\mathcal{V}$ に属する A 上の正規言語全体を $\mathcal{V}(A)$ と書くとき：

1. $\mathcal{V}(A) \subseteq \mathrm{Reg}(A)$ は部分ブール代数；
2. 任意の $L \in \mathcal{V}(A)$ と $w \in A^*$ に対し、$w\backslash L \in \mathcal{V}(A)$ および $L/w \in \mathcal{V}(A)$；
3. 任意の $L \in \mathcal{V}(B)$ とモノイド準同型 $f : A^* \to B^*$ に対し、$f^{-1}(L) \in \mathcal{V}(A)$.

を満たすときを言う。ただし、$w\backslash L$ と L/w はそれぞれ：

$$w\backslash L := \{u \in A^* \mid wu \in L\}$$
$$L/w := \{u \in A^* \mid uw \in L\}$$

により定義される正規言語で、それぞれ L の w による**左・右からの商**(quotient) という。

例えば $\mathcal{SF}$ を「star-free 言語の族」とすると、$\mathcal{SF}$ は上記の意味で正規言語の variety となっている。他にも $\mathcal{V}$ を「locally testable 言語の族」や「piecewise testable 言語の族」としても、やはり正規言語の variety であることが分かる。

「star-free 言語の族は variety である」とは、言い換えると「star-free であるという正規言語の性質は、上記 1〜3 の閉包性を持つ」ということだ。Eilenberg は、正規言語の性質 P が上記の閉包性を持つ時、与えられた正規言語 L がその性質 P を持つか否かを、syntactic monoid $M(L)$ の代数的性質で特徴づけられることを示した。

3.3 有限モノイドの Variety

Eilenberg が示したことをより厳密に理解するには、「syntactic monoid の代数的性質」の意味を正確にしなくてはならない。その際に中心となるのが、**有限モノイドの variety** という概念である。

定義 8. 有限モノイドの族 $\mathbb{V}$ が **variety** であるとは：

1. 任意の $M \in \mathbb{V}$ と部分モノイド $M' \leq M$ に対し、$M' \in \mathbb{V}$；
2. 任意の $M \in \mathbb{V}$ とその商 $M \twoheadrightarrow M'$ に対し、$M' \in \mathbb{V}$；
3. 任意の有限個の $M_i \in \mathbb{V}$ $(i = 1, \cdots, n)$ に対し、$\prod_i M_i \in \mathbb{V}$.

ただし、$\prod M_i$ は M_i 達の直積モノイドを表す。

たとえば、「aperiodic な有限モノイドの族」は有限モノイドの variety となる。また「全ての元が可逆な monoid (つまり群) の族」も有限モノイドの variety になる。

有限モノイドの variety に関する最も重要な事実の一つに、「有限モノイドの variety は常に**擬等式** (pseudo identity) と呼ばれる一種の等式で定義できる」というものがある。このことは例で見るのが一番分かりやすい。例えば aperiodic な有限モノイドの族 $\mathbb{A}$ は variety であったが、有限モノイド M に対し、$M \in \mathbb{A}$ であることと、M の全ての元 $x \in M$ が

以下の等式を満たすことは同値であることが知られている：

$$x^\omega = x^{\omega+1}. \tag{4}$$

ただし、有限モノイド M とその元 $x \in M$ に対し $x^\omega \in M$ とは、x^n なる元で idempotent であるものを表す。

上の式 (4) は擬等式の一つの例であって、上記の事実を「variety $\mathbb{A}$ は擬等式 $x^\omega = x^{\omega+1}$ で定義できる」という。簡単のため擬等式の一般的定義は Reiterman による原論文 [11] に譲るが、同様のことが有限モノイドの任意の variety について言える。つまり：

定理 2 (Reiterman). 有限モノイドの族 $\mathbb{V}$ が variety であることの必要十分条件は、それが擬等式の集合で定義可能であるときである。

以下では、有限モノイドの variety $\mathbb{V}$ が副有限等式の集合 E で定義できるとき、$\mathbb{V} = \mathbb{V}(E)$ と書く。例えば、(4) で述べたことから、aperiodic なモノイドの成す variety は $\mathbb{V}(x^\omega = x^{\omega+1})$ と一致する。

3.4 Variety 同士の一対一対応

最後の variety(有限オートマトンの variety) に立ち入る前にまず、正規言語の variety と有限モノイドの variety が一対一に対応していることを確認しておく方がいい。その一対一対応がまさに「正規言語の性質が、その syntactic monoid の性質で特徴づけられる」ことに相当している。

この対応を理解するにはまず、Schützenberger の定理を variety の言葉によって言い直すことが役に立つ。Schuützenberger の定理は、「正規言語 L が star-free であることと、その syntactic monoid $M(L)$ が aperiodic であることが同値」という物であった。これを $\mathcal{SF}$ と $\mathbb{A}$ を使ってそのまま言い換えると：

$$L \in \mathcal{SF} \quad \Leftrightarrow \quad M(L) \in \mathbb{A} \tag{5}$$

ということに他ならない。さらにもう一つ注意すべきことは、$\mathbb{A}$ は $M(L)$ $(L \in \mathcal{SF})$ なる有限モノイドを含む最小の (有限モノイドの)variety となっていることだ[5]。

一般に正規言語の variety $\mathcal{V}$ が与えられた時、$M(L)$ $(L \in \mathcal{V})$ なる有限モノイドを含む最小の (有限モノイドの)variety を $\mathcal{V}^\dagger$ と書くことにする。同様に、有限モノイドの variety $\mathbb{V}$ が与えられたとき、$M(L) \in \mathbb{V}$ なる正規言語 L 全体の族を $\mathbb{V}^\dagger$ と書く。するとこの時、$\mathbb{V}^\dagger$ は正規言語の variety であることが知られている。Eilenberg が示したのは、対応 $\mathcal{V} \mapsto \mathcal{V}^\dagger$ と $\mathbb{V} \mapsto \mathbb{V}^\dagger$ が互いに逆の全単射となることである。つまり：

定理 3 (Eilenberg). 正規言語の variety の全体の族を $\mathfrak{N}$、有限モノイドの variety の全体の族を $\mathfrak{M}$ とすると：

$$\mathfrak{N} \ni \mathcal{V} \mapsto \mathcal{V}^\dagger \in \mathfrak{M}$$
$$\mathfrak{M} \ni \mathbb{V} \mapsto \mathbb{V}^\dagger \in \mathfrak{N}$$

によって、$\mathfrak{N}$ と $\mathfrak{M}$ は同型。特に $\mathcal{V} \in \mathfrak{N}$ と $\mathbb{V} \in \mathfrak{M}$ に対して：

$$\mathcal{V} = \mathcal{V}^{\dagger\dagger} \tag{6}$$
$$\mathbb{V} = \mathbb{V}^{\dagger\dagger} \tag{7}$$

が成り立つ。

この結果と Reiterman の結果 (定理 2) を併せると、正規言語の variety $\mathcal{V}$ に対し「$L \in \mathcal{V}$ であるか否か」が、その syntactic monoid $M(L)$ の代数的性質によって特徴づけられるということの意味が分かる。上で言及している様に $\mathcal{V}^\dagger$ は有限モノイドの variety であるから、Reiterman の定理により、擬等式の集合 E が存在[6]して $\mathcal{V}^\dagger = \mathbb{V}(E)$ となる。また、$\mathcal{V} = \mathcal{V}^{\dagger\dagger} = \mathbb{V}(E)^\dagger$ であることと合わせると：

$$L \in \mathcal{V} \quad \Leftrightarrow \quad M(L) \text{ が } E \text{ を満たす}$$

ということが分かる。$M(L)$ が擬等式の集合 E を満たすか否かというのは、純粋に (モノイドに関する)

[5] そこまで自明ではない。

[6] 存在することは知られているが、一般にこの E を具体的に特定することはとても難しい。Schützenberger の最大の貢献は、$\mathcal{SF}^\dagger = \mathbb{V}(x^\omega = x^{\omega+1})$ であることを示したことであって、これは Eilenberg の定理の系として直ちに従う訳ではない。

代数的性質に他ならない。この意味で、「$L \in \mathcal{V}$ か否か」は $M(L)$ の代数的性質によって特徴づけられるということだ。

3.5 有限オートマトンの Variety

正規言語の性質は、その syntactic monoid の代数的性質と上記の意味で対応関係にあるが、それだけではない。正規言語の性質は、それを受理する有限オートマトンの幾何学的性質とも良い対応関係 [3] にあり、**有限オートマトンの variety** の概念によってそれが公理化される。

定義 9. 有限オートマトンの族 $\mathbf{V}$ が **variety** であるとは、次の条件を満たすときをいう：各アルファベット A に対して $\mathbf{V}(A)$ で、$\mathbf{V}$ に属する A 上の有限オートマトンの族を表すことにすると、

1. $\mathbf{V}(A)$ は自明なオートマトンを含む；
2. 任意の $\mathfrak{A} \in \mathbf{V}(A)$ と部分オートマトン $\mathfrak{A}' \leq \mathfrak{A}$ に対し、$\mathfrak{A}' \in \mathbf{V}(A)$；
3. 任意の $\mathfrak{A} \in \mathbf{V}(A)$ と商オートマトン $\mathfrak{A} \twoheadrightarrow \mathfrak{A}'$ に対し、$\mathfrak{A}' \in \mathbf{V}(A)$；
4. 任意の有限個の $\mathfrak{A}_i \in \mathbf{V}(A)$ に対し、$\prod_i \mathfrak{A}_i \in \mathbf{V}(A)$；
5. 任意の有限個の $\mathfrak{A}_i \in \mathbf{V}(A)$ に対し、$\coprod_i \mathfrak{A}_i \in \mathbf{V}(A)$；
6. 任意の $\mathfrak{A} \in \mathbf{V}(B)$ とモノイド準同型 $f : A^* \to B^*$ に対し、$f^{-1}\mathfrak{A} \in \mathbf{V}(A)$.

ただし、$\prod_i \mathfrak{A}_i$ と $\coprod_i \mathfrak{A}_i$ は、それぞれ $\mathfrak{A}_i$ 達の積および離散和オートマトンを表す。また、$\mathfrak{A} \in \mathbf{V}(B)$ を (Q, δ) とすると、$f^{-1}\mathfrak{A}$ は $(Q, f^{-1}\delta)$ によって与えられる。ここで、$f^{-1}\delta : Q \times A \to Q$ は：

$$q \xrightarrow{a} q' \quad \Leftrightarrow \quad q \xrightarrow{f(a)} q' \quad (a \in A)$$

となるように定義される。

正規言語の variety と有限オートマトンの variety は、正規言語 L に対しその syntactic monoid $M(L)$ を取ることで互いに対応していた。同様に、有限オートマトンの variety もこの対応関係に自然に加えることが出来る。

今、$\mathbf{V}$ を有限オートマトンの variety とすると、$\mathbf{V}$ に属すオートマトンによって受理される正規言語の族 $\mathbf{V}^\ddagger = \bigcup \mathbf{V}(A)$ が定義できる：

$$\mathbf{V}^\ddagger(A) := \{L \subseteq A^* \mid \exists \mathfrak{A} \in \mathbf{V}.\ L = L(\mathfrak{A})\}. \qquad (8)$$

すると $\mathbf{V}^\ddagger$ は、正規言語の variety になることが容易に分かる。一方、$\mathcal{V}$ を正規言語の variety とする。このとき $\mathcal{V}^\ddagger$ によって「$\mathcal{V}$ の言語を受理する有限オートマトンの族」と定義すると、$\mathcal{V}^\ddagger$ は有限オートマトンの variety となる。

定理 4. 有限オートマトンの variety 全体の族を $\mathfrak{F}$ とすると：

$$\mathfrak{A} \ni \mathcal{V} \mapsto \mathcal{V}^\ddagger \in \mathfrak{F} \qquad (9)$$

$$\mathfrak{F} \ni \mathbf{V} \mapsto \mathbf{V}^\ddagger \in \mathfrak{A} \qquad (10)$$

は互いに逆の、$\mathfrak{A}$ と $\mathfrak{F}$ の間の同型となる。特に $\mathcal{V} \in \mathfrak{A}$ と $\mathbf{V} \in \mathfrak{F}$ に対して：

$$\mathcal{V} = \mathcal{V}^{\ddagger\ddagger} \qquad (11)$$

$$\mathbf{V} = \mathbf{V}^{\ddagger\ddagger} \qquad (12)$$

が成り立つ。

正規言語の性質は、有限モノイドの性質および有限オートマトンの性質と密接に関連している。特に正規言語の組合せ的性質を有限モノイド・有限オートマトンの性質で特徴づけることで、その決定可能性が示せる場合がある。

正規言語の variety theory は、三つの対象：

1. 正規言語の variety
2. 有限モノイドの variety
3. 有限オートマトンの variety

間の一対一対応という形で、正規言語、有限モノイド、および有限オートマトンの間の関連性を体系化し、それにより正規言語の組合せ的性質の決定可能性を証明する、一つの強力なアプローチを与えてくれる。

4 Stone双対性による(再)証明

上述してきた正規言語のvariety theoryは、1965年のSchützenbergerの論文にさかのぼる長い歴史があるが、近年になってその再定式化・再証明が進みつつある。その発端となったのが1997年のPippenger[10]による研究で、その論文では、正規言語のvariety theoryにおける中心的結果の一端である、

1. 正規言語のvariety
2. 有限モノイドのvariety

の間の一対一対応を、**Stone双対定理**の系として再証明する試みがなされた。この研究は2008年にGehrke, Grigorieff およびPin[7]、Rhodes およびSteinberg[12]によって引き継がれ、その結果、上記二つのvariety間の対応については、その背景がより見通し良くなった[7]。

結論から言えば、正規言語のvarietyと有限モノイドのvarietyの間の対応は、

1. 正規言語が成す($\mathbb{F}_2$上の)**双代数**(bialgebra)
2. 有限モノイドの極限の**副有限モノイド**(profinite monoid)

の間のStone双対性からの直接の帰結として再証明出来る。とくにその再証明では、「正規言語のvarietyは、2元体$\mathbb{F}_2$上の双代数として特徴づけられる」という(RhodesとSteinbergによる)事実が鍵になる。尤もこの事実は、群の表現論で既に知られていた結果の別表現であって、彼らによる再証明は、群の表現論における幾つかの概念を使って、もう少し整理することが出来る。

[7]彼らの研究においてvarietyとは、アルファベットを固定したものを指しているが、本質的には違いはあまりない。本稿でもそれに従って、以下では、固定したアルファベット上の族$\mathcal{V}(A)$に制限したものをvarietyと呼ぶ。

4.1 表現関数と表現双代数

以下ではMを(位相)モノイド、kを体とする。一般にM上のk値の**表現関数**(representative function)[1]とは、Mの有限次元線形表現から以下のような方法で得られる(連続)関数を言う：

定義 10. 関数$f : M \to k$が**表現関数**(representative funciton)であるとは、Mの有限次元線形表現$\rho : M \to \mathrm{End}_k(V)$と線形関数$h : \mathrm{End}_k(V) \to k$が存在して、$f = h \circ \rho$と表せる時を言う。

この定義の代わりに、M上の表現関数とは、Mの有限次元線形表現の行列成分(matrix coefficient)の和として表される関数と言っても良い。$f : M \to k$を上記の通りとし、表現ρの表現空間Vの基底を一つ固定する。すると、Mの各元$s \in M$のρによる表現$\rho(s) \in \mathrm{End}_k(V)$は、正方行列$\rho(s) = \big(\rho_{i,j}(s)\big)$として表せる。また$E_{i,j} \in \mathrm{End}_k(V)$を$(i,j)$成分のみ1で他は0であるような行列とすれば、結果的に関数fは：

$$f = \sum_{i,j} h(E_{i,j}) \cdot \rho_{i,j}$$

という行列成分$\rho_{i,j}$の線形和で書ける。このこととρがモノイド準同型であることに注意すると、有限個の関数$f^i_{(1)}, f^i_{(2)} : M \to k$ $(i = 1, \cdots, n)$が存在して、任意の$s, t \in M$に対して次が成り立つことが容易に分かる：

$$f(s \cdot t) = \sum_{i=1}^{n} f^i_{(1)}(s) \cdot f^i_{(2)}(t). \tag{13}$$

逆に関数fに対して、このような等式が成り立つ$f^i_{(1)}, f^i_{(2)}$が存在するとき、fは表現関数であることが知られている。

さらに、M上のk値表現関数の全体を$R_k(M)$と置く時、$R_k(M)$には自然に**双代数**(bialgebra)の構造が入る。実際、$f \in R_k(M)$を表現関数とすると、上記したように式(13)を満たす関数$f^i_{(1)}, f^i_{(2)}$が存在す

るが、実はそれら全て表現関数となるようにとれる。この時、f の**余積**(comultiplication) $\Delta f \in R_k(M) \otimes R_k(M)$ を：

$$\Delta f := \sum_{i=1}^{n} f^i_{(1)} \otimes f^i_{(2)}$$

と定義することにより、$R_k(M)$ は**余代数**(coalgebra)の構造が入る。また、積 $\mu : R_k(M) \otimes R_k(M) \ni f \otimes g \mapsto f \cdot g \in R_k(M)$ は単純に：

$$(f \cdot g)(s) := f(s) \cdot g(s) \quad (s \in M)$$

によって定義できる。結果的に、この積 μ と余積 Δ によって、$R_k(M)$ が双代数になり、この双代数を一般に、M の**表現双代数**(representative bialgebra) と言う。

4.2 正規言語のなす双代数

Rhodes と Steinberg[12] はその著書の中で、「正規言語の成すブール代数 $\mathrm{Reg}(A)$ には、双代数の構造が自然に入る」ということを報告している。実はこの双代数は、自由モノイド A^* 上の表現双代数 $R_{\mathbb{F}_2}(A^*)$ に他ならないが、重要なのは、正規言語の variety をこの双代数 $\mathrm{Reg}(A)$ の部分双代数として特徴づけられるという点にある。

そもそも正規言語の双代数 $\mathrm{Reg}(A)$ と表現双代数 $R_{\mathbb{F}_2}(A^*)$ が同一視できるのは、言語 L が正規であることとその特性関数[8] $\hat{L} : A^* \to \{0,1\}$ が $\mathbb{F}_2$ 値の表現関数であることが同値になることによる。

命題 2. 言語 $L \subseteq A^*$ が正規言語である為の必要十分条件は、特性関数 $\hat{L} : A^* \to \mathbb{F}_2$ が表現関数となることである。

Rhodes と Steinberg は $\mathrm{Reg}(A)$ 上の双代数の構造を、正規言語の syntactic monoid を使った方法で定義しているが、それは表現双代数 $R_{\mathbb{F}_2}(A^*)$ のそれに一致している。とくに正規言語の variety を考える上で $\mathrm{Reg}(A)$ の双代数の構造が重要になるのは、正規言語の variety を双代数 $\mathrm{Reg}(A)$ の**部分双代数**として特徴付けられることによる。すなわち：

命題 3 (Rhodes-Steinberg). ブール部分代数 $\mathcal{V} \subseteq \mathrm{Reg}(A)$ が variety であるための必要十分条件は、$\mathcal{V}$ が双代数 $\mathrm{Reg}(A)$ の部分双代数となることである。

一方、有限モノイドの成す variety も別の等価なもので置き換えることが出来る。それは、**自由副有限モノイド**(free profinite monoid)[9] $\hat{A^*}$ の商 (quotient) となる副有限モノイド $\hat{A^*} \twoheadrightarrow M$ で、これは variety 内の有限モノイドの逆極限として得られる。逆にそのような副有限モノイド M から、有限モノイドへの全射 (finite quotient) 全体を考えることで、もとの variety が得られる。

4.3 正規言語と有限モノイドの Variety

正規言語の variety と有限モノイドの variety との間の一対一対応は、それぞれを「$\mathrm{Reg}(A)$ の部分双代数」および「自由副有限モノイド $\hat{A^*}$ の商 (quotient)」と取り替えることによって、Stone 双対定理の系として再証明出来る。その最後のひとステップとして必要な事実は、「$\mathrm{Reg}(A)$ の Stone 双対が $\hat{A^*}$ となること」という事実である。

一般に位相モノイド M に対して、その表現双代数 $R_{\mathbb{F}_2}(M)$ はブール環になっていて、従ってブール代数でもある。特にブール代数として $R_{\mathbb{F}_2}(M)$ の Stone 双対を考えることが出来るが、これは実は元のモノイド M の副有限完備化になっているということが比較的容易に分かる：

命題 4. 位相モノイド M に対し、その表現双代数 $R_{\mathbb{F}_2}(M)$ の Stone 双対は、M の副有限完備化 (profinite completion) $\hat{M}$ に同型である。特に、M が副有限モノイドの時、$R_{\mathbb{F}_2}(M)$ の Stone 双対は M 自身に同型になる。

[8] 一般に集合 X の部分集合 $L \subseteq X$ に対し、その特性関数 $\hat{L} : X \to \{0,1\}$ とは、$x \in L$ に対しては 1 を、$x \notin L$ に対しては 0 を対応させる関数のことをいう。

[9] 自由副有限モノイドとは、自由モノイド A^* の副有限完備化で得られる副有限モノイドのこと。また、モノイド M の**副有限完備化**(profinite completion) とは、M の有限商 $M \twoheadrightarrow H$ が成す系の逆極限のこと。

とくに $M = A^*$ と置けば、$\mathrm{Reg}(A)$ の Stone 双対が $\hat{A}^*$ となることが分かる。

以上の準備でようやく、正規言語の variety と有限モノイドの variety の間の一対一対応を、Stone 双対定理を経由して再証明することが出来るようになる。Stone 双対定理は、ブール代数の成す圏と Stone 空間 (コンパクト・ハウスドルフかつ完全不連結な位相空間) の成す圏の間の反変同値であったが、このことと「$\mathrm{Reg}(A)$ の Stone 双対は $\hat{A}^*$ である」という事実から：

1. $\mathrm{Reg}(A)$ の部分双代数 $\mathcal{V} \subseteq \mathrm{Reg}(A)$；
2. $\hat{A}^*$ の商である副有限モノイド $\hat{A}^* \twoheadrightarrow M$

とが一対一に対応していることが直ちに従う。一方、前節でこれらのそれぞれが、正規言語の variety と有限モノイドの variety と等価であることをみたが、結果的に 1. および 2. の一対一対応は、正規言語の variety と有限モノイドの variety の間の一対一対応を導くということが分かる。

5 擬ガロア圏と表現定理

正規言語の variety theory の主結果には、(1) 正規言語の variety と (2) 有限モノイドの variety 間の対応だけでなく、(3) 有限オートマトンの variety との対応も含まれる。前節では、(1) と (2) の対応を、(1') 双代数 $\mathrm{Reg}(A)$ の部分双代数と (2') 自由副有限モノイド $\hat{A}^*$ の商副有限モノイドの対応から再証明できることを観たが、ここではそれに (3) に相当するものを加える。

その為に、双代数 $\mathrm{Reg}(A)$ と副有限モノイド $\hat{A}^*$ が Stone 双対であるという事実は、それぞれの余加群・表現の成す圏の (反変) 同値性で置き換えられるという点に注意する。というのも、これらの圏は、ちょうど有限オートマトンを対象 (object) に持つ圏であって、$\mathrm{Reg}(A)$(の部分双代数) と $\hat{A}^*$(の商) の対応の代わりに余加群・表現の圏の反変同値性を考えることで、同時に有限オートマトン (の成す variety) も含めることができるからだ。

この節の目標は、有限オートマトンの variety を対象にもつような圏を、$\mathrm{Reg}(A)$ 余加群・$\hat{A}^*$ の表現という特定の表現に依らない形で特徴付けることにある。実際、それはちょうど、「正規言語の variety」を $\mathrm{Reg}(A)$ の双代数として特徴付けたことの、「有限オートマトンの variety」の場合に相当するものと言える。

5.1 ガロア圏

副有限モノイドの表現の圏ではなく副有限「群」の表現の場合には、そのような特徴付けが知られている。つまり、以下の**ガロア圏**(Galois category) の公理を満たす圏は常に、副有限群の表現の圏（或はその Stone 双対である Hopf 代数の余加群の圏）と同値となる (例えば [15] を参照)。 以下、**fsets** と書いて有限集合と写像の圏を表す。

定義 11. ガロア圏(Galois category) とは、圏 $\mathcal{C}$ と関手 $F : \mathcal{C} \to \mathbf{fsets}$ の組 $(\mathcal{C}, F)$ であって、以下の公理を満たすものを言う：

C_0) $\mathcal{C}$ は始対象 (initial object) $\emptyset$ と終対象 (final object) 1 を持つ；

C_1) $\mathcal{C}$ は有限のファイバー積 (fibred product) を持つ；

C_2) $\mathcal{C}$ は有限の余直積 (coproduct) を持つ；

C_3) $\mathcal{C}$ の任意の射 $f : X \to Y$ に対して、以下の様な射の分解が存在する：

$$\begin{array}{ccc} X & \xrightarrow{\quad f \quad} & Y \\ & {\scriptstyle \pi} \searrow \quad \nearrow {\scriptstyle j} & \\ & Z & \end{array} \qquad (14)$$

ここで $\pi : X \twoheadrightarrow Z$ は strict epimorphism であり、$j : Z \hookrightarrow Y$ は単射 (monomorphism)。また、ある射 $j' : Z' \hookrightarrow Y$ が存在して、Y は Z と Z' の余直積 $Z \sqcup Z'$ となる;

C_4) 任意の対象 $X \in \mathcal{C}$ と X に作用する有限群 H に対して、普遍商 (universal quotient) $\pi : X \twoheadrightarrow X/H$ が存在し、π は strict epimorphism；

F_0) $F(\emptyset) = \emptyset$ および $F(1) = 1$;

F_1) F はファイバー積を保つ；

F_2) F は余直積を保つ；

F_3) F は strict epimorphisms を全射に、単射を単射に移す；

F_4) F は普遍商を保つ；

F_5) F は同型を反映 (reflect) する.

ガロア圏は常に副有限群の表現の圏と同値であって、また逆も然りという事実が知られている。

定理 5 (Grothendieck). $(\mathcal{C}, F)$ をガロア圏とする。このとき、関手 $F : \mathcal{C} \to \mathbf{fsets}$ の自然同型 (natural isomophism) $F \Rightarrow F$ の全体 $\mathrm{Aut}(F)$ には、副有限群の位相が自然に入り、$\mathcal{C}$ は副有限群 $\mathrm{Aut}(F)$ の表現の圏と同値となる。

5.2 擬ガロア圏

副有限群の表現の圏を特徴づけるガロア圏の公理は、それを適切に弱めることで、ちょうど副有限モノイドの表現の圏と同値になるように出来る。ここでは仮に、その公理を満たす圏を**擬ガロア圏**(pseudo Galois category) とでも呼ぶことにして、次のように定義する。

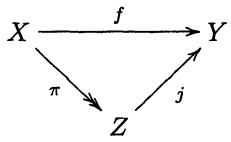
定義 12. 擬ガロア圏(pseudo Galois category) とは、圏 $\mathcal{C}$ と関手 $F : \mathcal{C} \to \mathbf{fsets}$ の組 $(\mathcal{C}, F)$ であって、以下の公理を満たすものを言う：

C_0) $\mathcal{C}$ は始対象 (initial object) $\emptyset$ と終対象 (final object) 1 を持つ；

C_1) $\mathcal{C}$ は有限のファイバー積 (fibred product) を持つ；

C_2) $\mathcal{C}$ は有限のプッシュアウト (pushout) を持つ；

C_3) $\mathcal{C}$ の任意の射 $f : X \to Y$ に対して、以下の様な射の分解が存在する：



(15)

ここで $\pi : X \twoheadrightarrow Z$ は strict epimorphism であり、$j : Z \hookrightarrow Y$ は単射 (monomorphism)。

F_0) $F(\emptyset) = \emptyset$ および $F(1) = 1$;

F_1) F はファイバー積を保つ；

F_2) F はプッシュアウトを保つ；

F_3) F は strict epimorphisms を全射に、単射を単射に移す；

F_4) F は同型を反映 (reflect) する.

勿論、ガロア圏はつねに擬ガロア圏になる。ガロア圏の公理にあって擬ガロア圏の公理に無いものは、例えば普遍商に関する公理や、単射 $Z \hookrightarrow Y$ があるとき Y が余直積 $Y = Z \sqcup Z'$ に分解するという公理などがある。

擬ガロア圏はもちろん、一般には副有限群の表現の圏とは同値にはならないが、代わりに副有限モノイドの表現の圏と同値になる。ガロア圏の場合と同様、その副有限モノイドは擬ガロア圏 $(\mathcal{C}, F)$ から具体的に構成できる。

定理 6. $(\mathcal{C}, F)$ を擬ガロア圏とする。このとき、関手 $F : \mathcal{C} \to \mathbf{fsets}$ の自然変換 $F \Rightarrow F$ 全体がなすモノイド $\mathrm{End}(F)$ には自然に副有限位相が入り、$\mathcal{C}$ は副有限モノイド $\mathrm{End}(F)$ の表現の圏と同値となる。

この事実と Stone 双対定理を使うと、(1) ブール代数上の双代数の圏、(2) 副有限モノイドの圏、および (3) 擬ガロア圏のなす圏[10]は互いに (反変) 同値となることも示せる。この三種の圏の間の同値性から殆ど直接の帰結として、正規言語、有限モノイド、および有限オートマトンの variety 間の一対一対応が示せる。

[10]擬ガロア圏の間の射は、ガロア圏の間の射 [15] と同様に定義される。

参考文献

[1] Eiichi Abe. *Hopf algebras.* Cambridge University Press, 2004.

[2] Janusz Brzozowski and Imre Simon. Characterizations of locally testable events. *Discrete Math.*, pages 243–271, 1973.

[3] Laura Chaubard, Jean-Eric Pin, and Howard Straubing. Actions, wreath products of c-varieties and concatenation product. *Theoret. Comput. Sci.*, pages 73–89, 2006.

[4] Samuel Eilenberg. *Automata, languages and machines. Vol. A.* Academic Press, 1974.

[5] Samuel Eilenberg. *Automata, languages and machines. Vol. B.* Academic Press, 1976.

[6] Jeffrey Friedl. *Mastering Regular Expressions.* Oreilly & Associates Inc., 2006.

[7] Mai Gehrke, Serge Grigorieff, and Jean-Eric Pin. Duality and equational theory of regular languages. In *ICALP 2008*, pages 246–257, 2008.

[8] Stephen Kleene. Representation of events in verve nets and finite automata. In *Automata studies*, pages 3–41, 1956.

[9] Jean-Eric Pin. *Varieties of formal languages.* Plenum Publishing Corp., 1986.

[10] Nicholas Pippenger. Regular languages and Stone duality. *Theory Comput. Syst.*, pages 121–134, 1997.

[11] Jan Reiterman. The birkhoff theorem for finite algebras. *Algebra Universalis*, pages 1–10, 1982.

[12] John Rhodes and Benjamin Steinberg. *The q-theory of finite semigroups.* Springer-Verlag, 2008.

[13] Marcel-Paul Schützenberger. On finite monoids having only trivial subgroups. *Informationa and Control*, pages 190–194, 1965.

[14] Imre Simon. Piecewise testable events. In *2nd GI Conf.*, pages 214–222, 1975.

[15] Fabio Tonini. Notes on Grothendieck-Galois theory, 2009.