Some Applications of the Theory of Semigroups to Automata

H. Jürgensen

Institut für theoretische Informatik
Technische Hochschule Darmstadt
Magdalenenstr. 11
6100 Darmstadt

In this paper we intend to give an impression of some applications of the theories of semigroups and groups to the theory of automata. In fact we shall not really restrict ourselves to automata but rather consider such applications to other related areas in theoretical computer science as well, i.e., in particular to languages and codes. In a survey like this one, which is supposed to introduce the non-specialist to the subject, it is of course neither possible nor desirable to cover the whole field. Instead, we shall rather present a few topics, which as we think are appropriate representatives of certain major directions of the area. Similarly the references to the literature we give are not meant to be complete; however, in most cases the papers referred to will themselves give hints concerning the development of and further literature on, the respective subject.

1. Automata

Automata have been defined as models of systems reacting upon sequences of stimuli. Computers are among the most common examples of automata. Living organisms, many types of machines, information transmission channels, complex systems as e.g. traffic are further typical examples.

An *automaton* A consists of

- the set $X \neq \emptyset$ of elementary stimuli, called *input symbols*,
- the memory S, considered as a nonempty set, the set of *states*, which may have an algebraical or topological structure,
- the set $Y \neq \emptyset$ of elementary reactions, called *output symbols*,

- the function $\delta: X\times S \rightarrow S\times Y$, which to each input symbol $x \in X$ and each state $s \in S$ associates by $\delta(x,s) = (s',y)$ the next state s' and the corresponding output symbol y.

It is a basic assumption in automaton theory, that an automaton works in discrete time and in a stationary manner. The sequential behaviour of $A = (X,S,Y,\delta)$ is given by extending δ to sequences of input symbols in the following natural manner: Let

$$X^{+} = \{x_1 x_2 \ldots x_n \mid n \in \mathbb{N},\ x_i \in X,\ i = 1,2,\ldots,n\}$$

be the set of all sequences of symbols in X, called *words* on X, let 1 denote the empty sequence, the *empty word*, and let

$$X^{*} = X^{+} \cup \{1\}.$$

For $x_1 \ldots x_n, y_1 \ldots y_m \in X^{*}$ the *concatenated* sequence $x_1 \ldots x_n y_1 \ldots y_m$ is again in X^{*}. X^{*} and X^{+} are semigroups with this multiplication. 1 is the identity element of X^{*}; hence X^{*} is a *monoid*. X^{+} is the *free semigroup*, X^{*} is the *free monoid on* X. Now

$$\delta: X^{*}\times S \rightarrow S\times Y^{*}$$

is defined as a partial mapping by

$$\forall s \in S:\ \delta(1,s) = (s,1)$$

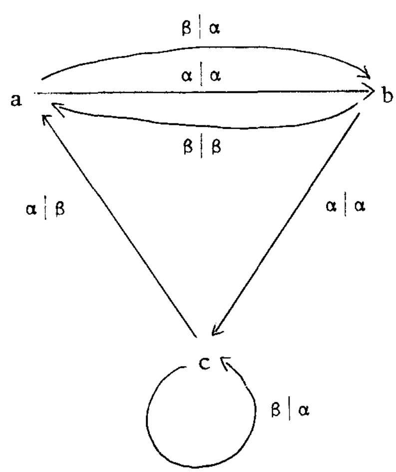
and

$$\forall w \in X^{*}\ \forall x \in X\ \forall s \in S\ \forall v \in Y^{*}\ \forall s' \in S:$$

$$(p_1(\delta(w,s))=s' \wedge p_2(\delta(w,s))=v) \longrightarrow \delta(wx,s)=(p_1(\delta(x,s')),vp_2(\delta(x,s'))),$$

where $p_1: S\times Y \rightarrow S$, $p_2: S\times Y^{*} \rightarrow Y^{*}$ are the projections.

Example: 1. Let $X = \{\alpha,\beta\}$, $S = \{a,b,c\}$, $Y = \{\alpha,\beta\}$. δ is visualized by a graph:



An arrow from s to s' with inscription $x|y$ says $\delta(x,s)=(s',y)$. The behaviour of the automaton can be read off this diagramme: Given e.g. the input sequence $\alpha\alpha\beta\beta\alpha\alpha\alpha\alpha\beta\beta$ in state b it would reach states c,a,b,a,b,c,a,b,a,b and produce $\alpha\beta\alpha\beta\alpha\alpha\beta\alpha\beta\alpha$ as the corresponding output sequence.

A theorem in automaton theory states that at the cost of some additional states the function δ may be represented by two functions δ' and μ, such that a step of the automaton consists of
- first going into the next state according to δ' and depending on the present state and the input symbol,
- then producing the output symbol according to μ and depending on the new state.

For this reason it is often possible and useful to neglect the outputs of an automaton.

Definition 1. Let K be a category. A *K-semiautomaton* $A = (X,S,\delta)$ consists of
- the set $X \neq \emptyset$ of *input symbols*,
- the object $S \in |K|$, the *state space*,
- the mapping $\delta : X \to \mathrm{Endo}(S)$, which to each $x \in X$ associates the *state transition* δ_x, an endomorphism of S.

A is *X-finite*, if X is finite.

Throughout this paper X-finiteness is assumed. The semiautomata that are being considered in most of the rest of the paper are *finite state semiautomata*, i.e., X-finite K-semiautomata with K the category of finite sets. Other important examples are *finite linear automata* with K the category of finite dimensional linear spaces [R1], *topological semiautomata* with K the category of topological spaces [B1], and *finite stochastic semiautomata* [Pa1]; in the latter case the objects are the sets of finite dimensional stochastic row vectors, the morphisms are given by stochastic matrices.

Let $A = (X,S,\delta)$ be a K-semiautomaton. The sequential behaviour of A extends δ to a homomorphism

$$\tau : X^* \to \mathrm{Endo}(S) : w \mapsto \delta_w,$$

where

$$\delta_1 = \mathrm{id}_S \text{ and } \delta_{xw} = \delta_x \delta_w$$

(writing morphisms on the right hand side). For w, w' X^* $\tau(w) = \tau(w')$, if and only if w, w' cause the same state transition of A.

<u>Definition 2.</u> $T(A) = \tau(X^*)$ is the *transition monoid* of A.

Of course, $\tau(X)$ is a set of generators of T(A). So T(A) is a finitely generated monoid.

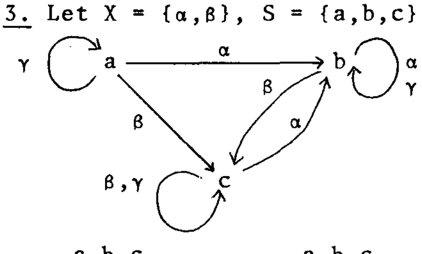
For many problems concerning semiautomata studying the structure of a semiautomaton A can thus be reduced to studying the structure of T(A), i.e., a subsemigroup of the endomorphism semigroup of its state space. Depending on K, Endo(S) for $S \in |K|$ is a semigroup whose structure may be reasonably well understood: This is e.g. true, if S is a free algebra [Du1]; in particular, a set S is a free algebra with no operations, and Endo(S) is the full semigroup of transformations of S, the *symmetric semigroup* on S, in this case; also vector spaces are useful examples in this respect; for a finite stochastic semiautomaton Endo(S) is the semigroup of all square stochastic matrices of a given dimension [Sw1].

<u>Example 2.</u> Let X, S, δ as in example 1, neglecting the outputs. Then

$$\delta_\alpha = \begin{pmatrix} a & b & c \\ b & c & a \end{pmatrix} \quad \text{and} \quad \delta_\beta = \begin{pmatrix} a & b & c \\ b & a & c \end{pmatrix}$$

are permutations generating the cyclic groups Z_3 and Z_2, respectively. T(A) being generated by these permutations is a group and thus a subgroup of the symmetric group S_S Therefore $T(A) = S_S$. A K-semiautomaton $A = (X, S, \delta)$ such that $T(A) \subseteq \text{Aut}(S)$, the automorphism group of S, is a *permutation K-semiautomaton*. If in addition T(A) is a group, A is a *group K-semiautomaton*. Of course, for a set $\text{Aut}(S) = S_S$, and a finite state permutation semiautomaton is a group semiautomaton.

<u>Example 3.</u> Let $X = \{\alpha, \beta\}$, $S = \{a, b, c\}$, δ as follows:



Then

$$\delta_\alpha = \begin{pmatrix} a & b & c \\ b & b & b \end{pmatrix} \qquad \delta_\beta = \begin{pmatrix} a & b & c \\ c & c & c \end{pmatrix}$$

generate a semigroup of order 2, and

$$\delta_\gamma = \binom{a\ b\ c}{a\ b\ c}$$

is the identity. Hence the multiplication of T(A) is given by

	δ_α	δ_β	δ_γ
δ_α	δ_α	δ_β	δ_α
δ_β	δ_α	δ_β	δ_β
δ_γ	δ_α	δ_β	δ_γ

A finite state semiautomaton like this one, such that for each $x \in X$ $\delta_x = id_S$ or $|S\delta_x| = 1$ is called a *reset semiautomaton.*

It is trivial that each (finite) monoid is the transition monoid of a *K*-semiautomaton with *K* the category of (finite) sets; one simply takes the right regular representation of the monoid. However, for other *K* this need not be true. So the problem of characterizing the transition monoids of finite linear semiautomata (over finite fields) has been attacked several times; the most complete treatment of this question has now been given by Eichner [Ec1].

2. Decomposition of finite state automata

One of the first and most prominent applications of semigroups to automata was in the decomposition of finite state (semi-)automata. The problem was to find a set of elementary finite state semiautomata such that each finite state semiautomaton can be simulated by a cascade of the elementary ones. The connection to semigroups is given by the facts that

A simulates B, if and only if T(B) is the homomorphic image of a subsemigroup of T(A)

and

the transition monoid of the cascade of A and B is the wreath product $T(A) \wr T(B)$.

Let S and T be semigroups. One says that S *divides* T $(S|T)$, if S is the homomorphic image of a subsemigroup of T. Let $\phi : T \to \mathrm{Endo}(S)$ be a morphism. The *right semidirect product* of S and T over ϕ, denoted by $S \times_\phi T$, is the semigroup, whose carrier is $S \times T$ and whose multiplication is defined by

$$(s,t)(s',t') = (s\phi(t)(s'),tt').$$

Let M be a class of monoids, and let $\overline{M}$ be the smallest class of monoids such that

$M \subseteq \overline{M}$,

$T \in \overline{M} \wedge S|T \rightarrow S \in \overline{M}$,

$S,T \in \overline{M} \wedge (\phi : T \rightarrow \mathrm{Endo}(S)$ a morphism$) \rightarrow S \times_{\phi} T \in \overline{M}$.

M is said to be *closed*, if $M = \overline{M}$ For instance the class of all finite monoids is closed. Given a closed class M of monoids, a monoid M is M-*irreducible*, if

$M|S \times_{\phi} T \wedge S \in M \wedge T \in M \rightarrow M|S \vee M|T.$

We can now state the monoid version of the decomposition theorem by Krohn and Rhodes, which is the generalization of a wellknown decomposition theorem for groups.

<u>Theorem 1</u> [e.g. Kr1,A1,La2,Ei1B,Ho1,De1]. Let M be the class of all finite monoids. $M \in M$ is M-irreducible, if and only if M is a finite simple group, or

$M \simeq R_3 = \langle x,y \mid xy=y=yy,\ yx=x=xx\rangle$

or

$M \simeq U_2 = \langle x \mid xx=x\rangle$.

In particular, if G_M is the set of subgroups of M, then $M \in \overline{G_M \cup \{R_3\}}$.

Replacing semidirect products by wreath products in order to handle transformation semigroups rather than abstract semigroups one can express the automaton theoretic consequence of this theorem as follows:

<u>Corollary 1</u>. Each finite state semiautomaton A is simulated by a cascade of finite permutation semiautomata whose transition monoids are simple groups and of reset automata with at most 2 states. In addition the simple groups obtained divide the maximal subgroups of T(A).

Theorem 1 has been the starting point for at least two directions of research in this area:

(1) The number of components into which a semiautomaton is decomposed and thus the number of states of the simulating cascade may be extremely great. This restricts the applicability of the theorem for practical purposes (say in hardware design). It is therefore interesting to consider situations, when the decomposition does not have a similarly disappointing result [e.g. Di1,Di2].

(2) Simplifying things a bit one might say that the actions of an automaton are really determined by interactions of permutations and resets. Therefore the minimum number of these interactions necessary to define the behaviour of the automaton can be considered as measuring its complexity. Accordingly, the *group complexity* of a monoid S is the minimum number of groups G_i such that

$$S \mid A_0 \wr G_1 \wr A_1 \wr G_2 \wr A_2 \ldots G_n \wr A_n,$$

where $\wr$ denotes the wreath product of monoids, and each A_i is an *aperiodic* semigroup, i.e., a semigroup with only trivial subgroups.

Theorem 2. There are finite semigroups of arbitrary group complexity.

This means that given a finite cascade of finite permutation automata and reset automata it may well be impossible to change the cascade in such a way that all the permutation automata would be grouped together.

Abstracting from group complexity a general theory of complexity of finite semigroups has been developped, which apart from its value for the understanding of the structure of finite semigroups has an interpretation as a theory of complexity of finite state semiautomata [A1]. An account of the theory has recently been given in [Ei1B] by Tilson.

3. Languages, events

Definition 3. Let X be a nonempty finite set. A *language* on X is a subset of X^*.

A congruence ρ on X^* is said to *saturate* $L \subseteq X^*$, if L is a union of ρ-classes. The *syntactic congruence* of L, σ_L, is the coarsest congruence on X saturating L. This definition is motivated by the property

$$u \,\sigma_L\, v \iff \forall x,y \in X^*: xuy \in L \leftrightarrow xvy \in L,$$

i.e., in terms of linguistics, $u \,\sigma_L\, v$, if and only if u and v behave syntactically the same. Thus $[u]_{\sigma_L}$, the equivalence class of u modulo σ_L, may be thought of as representing the syntactic function of u.

Definition 4. $S(L) = X^*/\sigma_L$ is the *syntactic monoid* of S.

With the above interpretation in mind, S(L) for a language L might serve as a means for constructing a grammar L [e.g. No1]. We shall not dwell on this interesting aspect any more but rather return to automata.

Definition 5. Let $A = (X,S,\delta)$ be a K-semiautomaton with K the category of sets. Let $s_0 \in S$, $F \subseteq S$. Then $A' = (A,s_0,F) = (X,S,\delta,s_0,F)$ is an *acceptor (K-acceptor*, to be precise) with *initial state* s_0 and set of *final states* F. A' *accepts* the language

$$L(A') = \{w \mid w \in X^*: \delta(w,s_0) \in F\}.$$

A' is a *finite state acceptor* if A is a finite state automaton.

Now, if $A' = (X,S,\delta,s_0,F)$ is an acceptor, then for $w,w' \in X^*$ $\tau(w) = \tau(w')$ implies $[w]_\sigma = [w']_\sigma$ with $\sigma = \sigma_{L(A')}$. Hence S(L(A')) is a homomorphic image of T(A). On the other hand, if $L \subseteq X^*$, then

$$A(L) = (X,S(L),\delta,[1]_{\sigma_L},L/\sigma_L)$$

with

$$\delta: X \times S(L) \to S(L): (x,[w]_{\sigma_L}) \mapsto [wx]_{\sigma_L}$$

is an acceptor for L. Moreover $T(A(L)) \cong S(L)$.

Part of the relevance of syntactic semigroups becomes transparent from the following theorem, which is a modification of a theorem due to Kleene [Kl1].

Theorem 3. For a language $L \subseteq X^*$ there is a finite state acceptor accepting L, if and only if S(L) is finite. Such a language is called *rational* (or *regular*).

Much work has been devoted to the study of the correspondence between classes $\mathcal{L}$ of (rational) languages and classes $\mathcal{M}$ of monoids given by

$$L \in \mathcal{L} \to \exists M \in \mathcal{M} : M \cong S(L)$$
$$M \in \mathcal{M} \wedge M \cong S(L) \to L \in \mathcal{L}.$$

Whereas each monoid is isomorphic with the transition monoid of an automaton, there are monoids, which are not isomorphic with any syntactic monoid. For the above correspondence to be of any use, there should be a nice characterization of L and M in combinatorial and algebraical terms, respectively. Following a suggestion of Schützenberger's [Sc2] an elegant abstract formulation of this correspondence has been given by Eilenberg [Ei1B]. Here the M's are *(pseudo-)varieties of monoids*, i.e.,

$$\left.\begin{array}{l} M_1, M_2 \in \mathcal{M} \\ M \text{ a submonoid of } M_1 \times M_2 \\ \phi \text{ a morphism of } M \end{array}\right\} \rightarrow \phi(M) \in \mathcal{M},$$

and the L's are *varieties of languages*, i.e., defined by classes $L(X)$ of languages on X with X any nonempty finite set, such that $L(X)$ is closed under the Boolean operations, right and left division by symbols, and $f^{-1}(L(X)) \subseteq L(Y)$ for each Y and each morphism $f:Y^* \rightarrow X^*$. The conditions defining varieties of languages can be motivated from their automaton theoretic interpretation.

Theorem 4 [Ei1B]. For L and M varieties of rational languages and finite monoids, respectively, let $\mu(L)$ be the variety of monoids generated by $\{S(L) \mid \exists X:L \in L(X)\}$, and let $\lambda(M) = \{L \mid \exists M \in M : M \approx S(L)\}$. Then $\mu\lambda(M) = M$ and $\lambda\mu(L) = L$.

We mention one instance of this correspondence, which has a comparatively simple formulation:

Theorem 5 [Sc2,Mc1]. For each X let $L_H(X)$ be the class of languages on X accepted by finite state acceptors without nontrivial loops (i.e. no input symbol induces a permutation on a non-singleton set of states). Let M_H be the class of finite monoids whose H-classes are singleton sets (equivalently, whose subgroups are trivial). Then $\lambda(M_H) = L_H$, $\mu(L_H) = M_H$.

In [Mc1] several other and not quite as simple characterizations of L_H are given (also in [Sc2]). Theorem 5 has been extended to probabilistic events in [Jü1] with an appropriate generalization of definition 5 to arbitrary concrete categories. There is a similar corrrespondence between the varieties of piecewise testable languages and J-trivial finite monoids, i.e., finite monoids each of whose principal ideals has a unique generator [Si1]. The varieties of R-trivial or L-trivial finite monoids have been studied in [Br1,Br2, Br3]. A survey on varieties "below" M_H and L_H has been given by

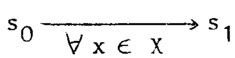
Brzozowski [Br4]. A deep theorem on these lines was recently proved by Schützenberger [Sc3]. Valkema [Va1,Va2] gave examples of naturally defined classes of rational languages and of finite monoids with the above correspondence such that neither is a variety. This seems to indicate that a weakened variety concept might be interesting to have.

For non-rational languages the study of the correspondence between languages and monoids is still in its infancy. Several interesting, though still sporadic results have been obtained by Perrot and Sakarovitch [Pe1,Pe2,Pe3,Sa1]. A slightly different line of research has been followed in [Jü2,Jü3,La3,Ma1], where the problem of finding all languages L such that $S(L) \cong M$ for a single monoid M in a given class of monoids is considered.

One important aspect of syntactic monoids is that they enable or simplify proofs of propositions in the theories of automata and languages. As an example we mention a recent theorem by Brzozowski and Knast [Br5] that solves the following automaton theoretic problem, which has been open for several years:

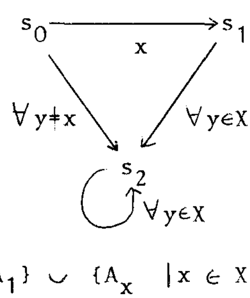
Let X be a finite nonempty set. Consider the finite state acceptors

A_1:



with $F = \{s_0\}$,

A_x:



with $F = \{s_1\}$ and x X.

Let $E_X = \{A_1\} \cup \{A_x \mid x \in X\}$. The acceptors in E_X are considered as the elementary components for building finite state acceptors using parallel or series composition or negation; for a finite state acceptor $A = (X,S,\delta,s_0,F)$ the *negation* is $\bar{A} = (X,S,\delta,s_0,S\setminus F)$, i.e., $L(A) = X^*\setminus L(A)$. Let $A_H(X)$ be the set of finite state acceptors with a composition like this, then

$$\{L \mid \exists A \in A_H(X) : L(A) = L\} = L_H(X).$$

Let $\mathcal{D}_n(X)$ be the set of $L \subseteq X^*$, such that there is $A \in A_H(X)$ accepting L, which contains no series longer than n. Clearly

$$\mathcal{D}_0(X) \subseteq \mathcal{D}_1(X) \subseteq \dots$$

and

$$\bigcup_{i \geq 0} \mathcal{D}_i(X) = L_H(X).$$

n may be considered as measuring the complexity of a definition of $L \in \mathcal{D}_n(X)$ in terms of series compositions (in fact, using regular expressions instead of automata, this becomes clearer). Therefore the sequence $\{\mathcal{D}_n(X)\}$ constitutes a description of $L_H(X)$ by increasing complexities. The question, whether there are languages of arbitrary complexity in $L_H(X)$ has been known as the dot-depth problem for several years. This has now been proved:

Theorem 6 [Br5]. For $|X| = 1$, $\mathcal{D}_0(X) \subsetneq \mathcal{D}_1(X) = L_H(X)$. For $|X| > 1$, $\mathcal{D}_0(X) \subsetneq \mathcal{D}_1(X) \subsetneq \dots \subsetneq \mathcal{D}_n(X) \subsetneq \mathcal{D}_{n+1}(X) \subsetneq \dots$

4. Codes

One of the most important applications of semigroup theory in theoretical computer science is in the theory of codes.

Definition 6. Let C,Y be nonempty finite sets. A *(uniquely decipherable) coding* of Y on X is a monomorphism $\phi: Y^* \to X^*$. $\phi(Y)$ is the corresponding *code*.

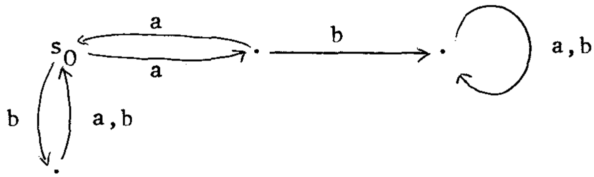
For $C \subseteq X^*$, let $\langle C \rangle$ be the submonoid of X^* generated by C. Evidently C is a (uniquely decipherable) code, if and only if $\langle C \rangle$ is freely generated by C. $\langle C \rangle$ may be considered as the set of coded words.

A code $C \subseteq X^*$ is said to be *decipherable without delay*, if

$$a \in C,\ v \in X^*,\ av \in \langle C \rangle \implies v \in \langle C \rangle.$$

Equivalently, $C \cap CXX^* = \emptyset$, i.e., no code word is the proper prefix of any other one. Hence these codes are called *prefix codes*. A (prefix) code C is *complete*, if there is no (prefix) code $C' \subseteq X^*$ properly containing C. A finite complete code, which is decipherable with bounded delay, is a prefix code [Sc1].

Example 5. Let $X = \{a,b\}$, $C = \{aa,ba,bb\}$. C is a prefix code. $\langle C \rangle$ is accepted by



with initial state s_0 and single final state s_0.

In the example $\langle C \rangle$ is the stabilizer of s_0 in X^*. This is not by chance; in fact:

Theorem 7 [e.g.Pe4]. $C \subseteq X^*$ is a finite prefix code, if and only if $\langle C \rangle$ is the stabilizer in X^* of the initial state of a finite state acceptor (X,S,δ,s_0,F). If in addition C is complete, then $\forall s \in \delta(X^*,s_0) \quad \exists w \in X^*: \delta(w,s) = s_0$.

This can be considered as one motivation for studying $S(\langle C \rangle)$ and $T(A(\langle C \rangle))$ for codes C. For instance:

Theorem 8 [Sc4]. If $C \subseteq X^*$ is a finite code such that $S(\langle C \rangle)$ is a group, then C is a block code of block length n for an $n \in \mathbb{N}$ and $S(\langle C \rangle)$ is isomorphic with the cyclic group Z_n.

A code C is said to be *synchronizing*, if there is a word $w \in X^*$ with $X^*w \subseteq \langle C \rangle$. This property is useful in decoding long words, where the presence of w, say in the form uwv, allows one to decode uw and v separately.

Theorem 9 [Sc5]. A finite complete prefix code C is synchronizing, if and only if the minimal ideal $I(\langle C \rangle)$ of $S(\langle C \rangle)$ is $\mathcal{H}$-trivial, i.e., if the maximal subgroups of $I(\langle C \rangle)$ are trivial groups.

For a finite complete prefix code C let $G(\langle C \rangle)$ be the isomorphism type of the maximal subgroups of $I(\langle C \rangle)$; $G(\langle C \rangle)$ or, to be precise, a particular isomorphic permutation group, the Suschkewitsch group of C, has been studied by Perrot [Pe4,Pe5] and Perrin [P1]. It was proved that $G(\langle C \rangle)$ cannot be arbitrary for finite C, for instance:

Theorem 10 [Pe4,Pe5]. For a finite complete prefix code C the centre of G(<C>) is cyclic (or trivial).

Further properties of the Suschkewitsch group of finite prefix codes have been proved in the above papers by Perrot and Perrin. It is interesting to see that in particular certain simple groups, e.g. the Mathieu groups, turn up with new representations as Suschkewitsch groups of finite prefix codes.

What is extremely important about the Suschkewitsch group is the fact, that to a certain extent its structure reflects structure properties of the code and its acceptor. We shall restrict to illustrating the former fact.

Consider codings $\phi: Y^* \to Z^*$, $\psi: Z^* \to X^*$ with $C' = \phi(Y)$ and $C'' = \psi(Z)$. Then $\chi = \psi\phi: Y^* \to X^*$ is again a coding, and $C = \chi(Y)$ is the *composition* of C' and C'', $C = C' \otimes C''$. (For linear block codes this would be the usual tensor product). There is a strong relation between compositions of prefix codes and of the Suschkewitsch groups [Pe4,Pe5], for instance:

Theorem 11. The Suschkewitsch group G of a finite complete prefix code C is a direct product $G \simeq G_1 \times G_2$ with $|G_1|$, $|G_2|$ relatively prime, if and only if C has two decompositions $C = C_1 \otimes C_2 = C_1' \otimes C_2'$ with $G(C_1) \simeq G(C_1') \simeq G_1$ and $G(C_2) \simeq G(C_2') \simeq G_2$.

References

A1 Arbib, M.A. (ed.): Algebraic theory of machines, languages, and semigroups. New York, 1968.

B1 Brauer, W.: Zu den Grundlagen einer Theorie topologischer sequentieller Systeme und Automaten. GMD, Bericht 31, Bonn, 1970.

Br1 Brzozowski, J.A., K. Culik II, and A. Gabrielian:Classification of non-counting events. J.Comp.Syst.Sci. 5 (1971), 41-53.

Br2 Brzozowski, J.A.: Run languages. Seminarbericht 87, Inst. f. Rechner- u. Programmstrukturen, GMD, 1975.

Br3 Brzozowski, J.A.: On aperiodic I-monoids. Res. report CS-75-28, Univ. of Waterloo.

Br4 Brzozowski, J.A.: Hierarchies of aperiodic languages. RAIRO, Inf. thêor. 10 (1976), 33-49.

Br5 Brzozowski, J.A., and R. Knast: The dot depth hierarchy of star free languages is infinite. Res. report CS-76-23, Univ. of Waterloo.

De1 Deussen, P.: Krohn-Rhodes-decomposition revised universally. Fak. f. Informatik, Univ. Karlsruhe, Bericht 9, 1975.

De2 Deussen, P.: Halbgruppen und Automaten. Berlin, 1971.

Di1 Dilger, E.: On permutation-reset automata. Inform. and Contr. 30 (1976), 86-95.

Di2 Dilger, E.: Zur Strukturtheorie abstrakter Automaten. Diss. Tübingen, 1977.

Du1 Dubreil, P.: Endomorphismes. Sêm. Dubreil-Pisot (algêbre et thêor. des nombres) 18 (1964/65), No.23.

Ec1 Eichner, L.: Die Übergangshalbgruppen linear realisierbarer endlicher Automaten. To appear.

Ei1 Eilenberg, S.: Automata, languages, and machines, A,B. New York, 1974, 1976.

G1 Gêcseg, F., and I. Peâk: Algebraic theory of automata. Budapest, 1972.

Ho1 Hotzel, E.: Der Kaskadenzerlegungssatz für Halbautomaten. In: GI, 1. Fachtagung über Automatentheorie und formale Sprachen, Bonn, 1973 (K.-H. Böhling and K. Indermark, eds.), Berlin, 1973 (= Lect. Notes in Comp. Sci. 2), 54-63.

Jo1 Johansen, P.: An algebraic normal form for regular events. Lyngby, 1972.

Jü1 Jürgensen, H.: Einige Eigenschaften der Monoide von Akzeptoren und Ereignissen. EIK 11 (1975), 545-556.

JÜ2 Jürgensen, H.: Syntaktische Halbgruppen. Proc. 1st workshop-meeting on categorical and algebraic methods in computer science and system theory, 1976. Bericht 37, Abt. Informatik, Univ. Dortmund, 1976, 37-43.

Jü3 Jürgensen, H.: Inf-Halbverbände als syntaktische Halbgruppen. To appear in Acta Math. Acad. Sci. Hung.

Kl1 Kleene, S.C.: Introduction to metamathematics. Amsterdam, 1952.

Kr1 Krohn, K., and J.L. Rhodes: Algebraic theory of machines, I. Prime decomposition theorem for finite semigroups and machines. Trans.AMS 116 (1965), 450-464.

La1 Lallement, G.: The rôle of regular and inverse semigroups in the theory of finite state machines and languages. Proc. sympos. on inverse semigroups, De Kalb, 1973, 44-63.

La2 Lallement, G.: On the prime decomposition theorem for finite monoids. Math. Syst. Theor. 3 (1969), 110-118.

La3 Lallement, G., and E. Milito: Recognizable languages and finite semilattices of groups. Semigroup Forum 11 (1975), 181-184.

M1 Марков, А.А.: Нерекуррентое кодирование. Проблемы кибернетики 8 (1962), 168-186.

Ma1 Maxson, C.J., and M. Zeller: Elementary Γ-languages and finite semilattices of groups. Semigroup Forum 13 (1976/77), 385-386.

Mc1 McNaughton, R., and S.Papert: Counter-free automata. Cambr., Mass., 1971.

No1 Novotný, M.: Einführung in die algebraische Linguistik. Rhein.-Westf. Inst. f. Instrum. Math., Bonn, IIM-Skript 40/67, 1967.

Pa1 Paz, A.: Introduction to probabilistic automata. New York, 1971.

P1 Perrin, D.: Codes bipréfixes et groupes de permutations. Thèse, Paris, 1975.

Pe1 Perrot, J.-F.: Monoides syntactiques des langages algébriques. Acta Inform. 7 (1977), 399-413.

Pe2 Perrot, J.-F., and J. Sakarovitsch: A theory of syntactic monoids for context-free languages. Proc. IFIP, Toronto,1977.

Pe3 Perrot, J.-F.: Une théorie algébrique des automates finis monogènes. Sympos. Math. 15, 201-244.

Pe4 Perrot, J.-F.: Contribution à l'étude des monoides syntactiques et de certains groupes associés aux automates finis. Thèse, Paris, 1972.

Pe5 Perrot, J.-F.: Groupes de permutations associés aux codes préfixes finis. In: Permutations, Paris, 1972 (A. Lentin, ed.), Paris, 1974, 19-35.

Pe6 Perrot, J.-F.: Informatique et algèbre - la théorie des codes à longueur variable. In: Theoretical computer science, 3rd GI conference,Darmstadt, 1977 (H. Tzschach, H. Waldschmidt, H.K.-G. Walter, eds.), Berlin, 1977, 27-44.

R1 Reusch, B.: Lineare Automaten. Mannheim, 1969.

Sa1 Sakarovitch, J.: Monoides syntactiques et langages algébriques. Thèse 3ème cycle, Paris, 1976.

S1 Schein, B.M.: Stationary subsets, stabilizers, and transitive representations of semigroups. Dissertationes Math. (Rozprawy Mat.) 77 (1970).

Sc1 Schützenberger, M.P.: On a question concerning certain free submonoids. J. Comb. Theor. 1 (1966), 437-442.

Sc2 Schützenberger, M.P.: Sur les monoides finis n'ayant que des sous-groupes triviaux. Sém. Dubreil-Pisot (alg. et théor. des nombres) 18 (1964/65), No. 10.

Sc3 Schützenberger, M.P.: Sur le produit de concaténation non ambigu. Semigroup Forum 13 (1976), 47-75.

Sc4 Schützenberger, M.P.: Une théorie algébrique du codage. Sém. Dubreil-Pisot (algèbre et théor. des nombres) 1955/56, No. 15.

Sc5 Schützenberger, M.P.: On the synchronizing properties of certain prefix codes. Inform. and control 7 (1964), 23-36.

Sw1 Schwarz, St.: On the structure of the semigroup of stochastic matrices. Publ.Math.Inst.Hung. Acad. Sci. A 9,3 (1964), 297-311.

Si1 Simon, I.: Piecewise testable events. In: GI, 2nd conf. on automata theory and formal languages, Kaiserslautern, 1975 (H. Brakhage, ed.), Berlin, 1975 (=Lect. Notes in Comp. Sci. 33), 214-222.

Va1 Valkema, E.: Zur Charakterisierung formaler Sprachen durch Halbgruppen. Diss., Kiel, 1974.

Va2 Valkema, E.: Syntaktische Monoide und Hypercodes. Semigroup Forum 13 (1976/77), 119-126.