THE UNIVERSITY OF MICHIGAN
COLLEGE OF LITERATURE, SCIENCE, AND THE ARTS
Department of Communication Sciences

Technical Report

DECOMPOSITIONS OF AUTOMATA USING NORMAL SUBMONOIDS

Bernard Zeigler

ORA Projects 08226 and 03105

supported by:

U.S. ARMY RESEARCH OFFICE (DURHAM)
CONTRACT NO. DA-31-124-ARO-D-483
DURHAM, NORTH CAROLINA

and

DEPARTMENT OF THE NAVY
OFFICE OF NAVAL RESEARCH
CONTRACT NO. Nonr-1224(21)
WASHINGTON D.C.

administered through:

OFFICE OF RESEARCH ADMINISTRATION ANN ARBOR

December 1967

RESEARCH PROGRESS REPORT

Title: "Decompositions of Automata Using Normal Submonoids," B. Zeigler, University of Michigan Technical Report 03105-50-T.

Background: The Logic of Computers Group of the Department of Communication Sciences of The University of Michigan is investigating the application of logic and mathematics to the design of computing automata. The application of the techniques and concepts of abstract algebra to automata forms a part of this investigation.

Condensed Report Contents: Aspects of decomposition of automata using normal submonoids are discussed. A simplified cascade connection procedure is given which can be specialized to the case of normal subgroups, but which introduces some novel features, e.g., the use of feedback in the tail machine, when applied to normal submonoids in general. Differences in efficiency between decompositions based on semi-direct products of machines versus those of semigroups are also discussed.

For Further Information: The complete report is available in the major Navy technical libraries and can be obtained from the Defense Documentation Center. A few copies are available for distribution by the author.

Section 1

Decompositions Based on Normal Submonoids

An automaton (or machine) is a triple $M = \langle X, Q, \circ\rangle$ where X is an (input) monoid; Q, a set (of states) and $\circ$, a map $(Q,X) \to Q$ (called the operation or action of X on Q) satisfying the conditions:

1. $q\circ 1 = q$ for all $q \in Q$, where 1 is the identity of X,

2. $q\circ(x_1x_2) = (q\circ x_1)\circ x_2$ for all $q \in Q$, $x_1, x_2 \in X$.

In the usual interpretation, X is a free monoid on a set of generators, S (the input alphabet) and the operation $\circ$ is obtained by extending to X, the transition function $(Q,S) \to Q$.[1] (As is customary we supress mention of $\circ$ where no ambiguity arises.)

The monoid, $X/\equiv_M$ of an automaton $M = \langle X, Q\rangle$ is the quotient monoid of X modulo the two-sided congruence:

$$x \equiv_M y \Longleftrightarrow q\circ x = q\circ y \text{ for all } q \in Q$$

i.e., $X/\equiv_M$ is the set of equivalence classes of $\equiv_M$ under multiplication $[x]\circ[y] = [xy]$.

A <u>monoid machine</u> is an automaton $M[X/\equiv] = \langle X, X/\equiv\rangle$ where $\equiv$ is a two-sided congruence on X and $[x]\circ y = [xy]$ for all $x,y \in X$. It is easily verified that the monoid of $M[X/\equiv]$ is precisely $X/\equiv$. (For more on the relation between monoid machines and "ordinary" machines see [2] and references therein.)

Now let K be a submonoid of X (by which we mean $1 \in K$ and $x,y \in K$ implies $xy \in K$). K is said to be <u>normal</u> if $xK = Kx$ for all $x \in X$. The equivalence

$$x \underset{K}{\equiv} y \text{ iff } Kx = Ky^2$$

---

[1] We use here the formulation given in Y. Give'on [1].

[2] $Kx = \{kx | k \in K\}$.

is already a right congruence (for $Kx = Ky$ implies $Kxz = Kyz$ for all $z \in X$) with equivalence classes $[x]$. Thus there is a machine

$$M[X/K] = \langle X, X/\equiv_K \rangle$$

with operation $[x] \circ y = [xy]$ for all $x, y \in X$.

It is easily demonstrated that if K is a normal submonoid of X, the relation $\equiv_K$ is a two sided congruence (see [3], Chapter 1 for a detailed account of semi-group congruences). Thus M[X/K] is a monoid machine if K is normal. In general $[x] \subsetneq Kx$ for all $x \in X$ (since $y \underset{K}{\equiv} x$ implies $y \in Ky = Kx$) and it is easily verified that if X is finite, $[x] = Kx$ (for all $x \in X$) just in case K is a subgroup of units (elements of X having inverses). In the event $[x] = Kx$ for all $x \in X$ the normality conditions can be weakened in the following sense: for M[X/K] to be a monoid machine it is necessary and sufficient that $xK \subseteq Kx$ for all $x \in X$. (Proof: use $[1] = K$ to establish that $[x][y] = [xy]$ iff $KxK = Kx$ iff $xK \subseteq Kx$).

That M[X/K] may be a monoid machine if $xK \subset Kx$ for some $x \in X$ is demonstrated in the monoid machine:

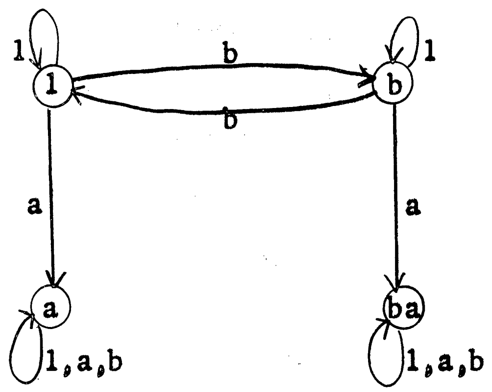


Fig. 1

The monoid is $X = \{a,b \mid a^2 = a,\ ab = a,\ b^2 = 1\}$ where $K = \{1,b\}$ is a subgroup. $X/\equiv_K = \{K, Ka\}$ where $Ka = Kba = \{a, ba\}$. Here $aK = \{a\}$, $baK = \{ba\}$ so both aK and baK are strictly included in Ka while $X/\equiv_K$ is multiplicative and M[X/K] a monoid machine.

In order to interconnect automata we shall have to supply them with outputs. Let $M = \langle X, Q \rangle$ and Y an (output) monoid. A map $Z : (Q,X) \to Y$ (in which we shall denote $Z(q,x) = x^q$ with the mnemonic that both x and x^q are strings and x^q is the output string resulting from input x in state q) is an admissible output map for M under the conditions:

1. $1^q = 1$ for all $q \in Q$ (the 1's refer to identities of X and Y respectively).
2. $(x_1x_2)^q = x_1^q x_2^{q \circ x_1}$ for all $q \in Q$, $x_1, x_2 \in X$.

The reader may wish to convince himself that an admissible map is a no memory (or one-state) device (if X, Y are free, condition 2 implies that Z is fixed once values have been assigned for the generators and for all $q \in Q$ and that Z is length preserving).

Now let $M_1 = \langle X, Q \rangle$ and $M_2 = \langle Y, R \rangle$ be two machines with $Z : (Q,X) \to Y$ an admissible map for M_1. The semi-direct product of M_1, M_2 via Z is a machine

$$M_1 \otimes_z M_2 = \langle X, Q \times R \rangle$$

with operation defined by

$$(q,r)x = (qx, rx^q)$$

for all $(q,r) \in (Q,R)$ and $x \in X$. It is easily verified that $M_1 \otimes_z M_2$ satisfies the well-definedness conditions given above.

Diagrammatically, $M_1 \otimes_z M_2$ is

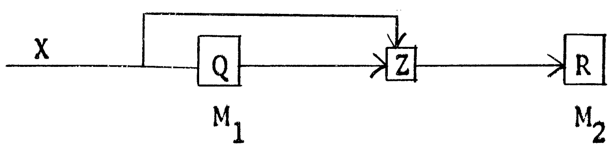


Fig. 2

In order to capture the apparently different configuration

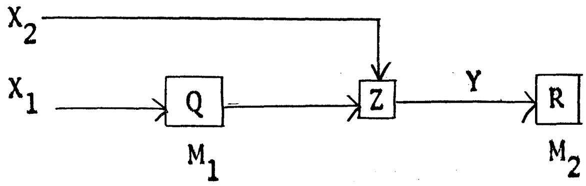


Fig. 3

we take X to be the submonoid of the product of free monoids X_1, X_2 such that

$$X = \{(x_1,x_2) | \ell(x_1) = \ell(x_2)\} \cup 1$$

where ℓ denotes length in terms of the generators. Make the operation of M_1 independent of X_2, i.e.,

$$q(x_1,x_2) = qx_1 \text{ for all } q \in Q, (x_1,x_2) \in X$$

and the output of Z independent of the present value of X_1, i.e.,

$$(x_1s_1,x_2s_2)^q = (x_1,x_2)^q s_2^{qx_1}$$

for all $(x_1,x_2) \in X$ and generators s_1 of X_1, s_2 of X_2.

Fig. 3 is the configuration inherent in semi-direct products of semigroups and, as we shall see, the independencies imposed may result in unnecessarily uneconomical decompositions when adapted to machines. The question of when is the semidirect product of monoid machines again a monoid machine bears on the relation between semidirect products of monoids and of machines.

<u>Proposition 1.</u> Let $M[X/\equiv_X]$, $M[Y/\equiv_Y]$ be monoid machines and $Z : (X/\equiv_X,X) \to Y/\equiv_Y$ an admissible map such that

$$u^{[1]} \equiv_Y v^{[1]} \text{ implies } u^{[x]} \equiv_Y v^{[x]}$$

for all x, u, v $\in$ X. Let $M_{1,1}$ denote the semi-direct product machine $M[X/\equiv_X] \otimes_Z M[Y/\equiv_Y]$ restricted to states accessible from ([1], [1]). Then $M_{1,1}$ is a monoid machine.

Proof. Define the right congruence on X

$$u \equiv v \text{ iff } ([1],[1])u = ([1],[1])v$$
$$\text{iff } u \equiv_X v \text{ and } u^{[1]} \equiv_Y v^{[1]}.$$

Then the map $([x],[x^{[1]}]) \to [x]$ is a machine isomorphism between $M_{1,1}$ and $M[X/\equiv]$. Showing that $\equiv$ is also a left congruence completes the proof since then $M[X/\equiv]$ is a monoid machine and by isomorphism so is $M_{1,1}$.

But left-sideness of $X/\equiv$ follows from the assumption:

$$u^{[1]} \underset{Y}{\equiv} v^{[1]} \text{ implies } u^{[x]} \underset{Y}{\equiv} v^{[x]}$$
$$\text{implies } x^{[1]}\, u^{[x]} \underset{Y}{\equiv} x^{[1]}\, v^{[x]}$$
$$\text{implies } (xu)^{[1]} \underset{Y}{\equiv} (xv)^{[1]}$$

So that for all $x \in X$

$$u \equiv v \text{ implies } u \underset{X}{\equiv} v \text{ and } u^{[1]} \underset{Y}{\equiv} v^{[1]}$$
$$\text{implies } xu \underset{X}{\equiv} xv \text{ and } (xu)^{[1]} \underset{Y}{\equiv} (xv)^{[1]}$$
$$\text{implies } xu \equiv xv$$

Q.E.D.

Now let X, Y be semi-groups and $Z : (X,Y) \to Y$ such that

1. $(y_1y_2)^x = y_1^x y_2^x$

2. $y^{x_1x_2} = (y^{x_2})^{x_1}$

for all $x, x_1, x_2 \in X$, $y, y_1, y_2 \in Y$. Then $X \circledast_Z Y$ is a semi-group semi-direct product where

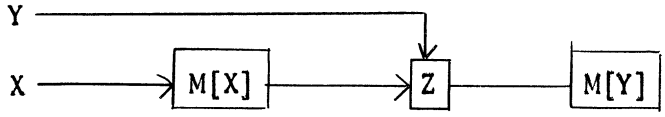
$$(x_1,y_1)(x_2,y_2) = (x_1x_2, y_1y_2^{x_1}) \quad .$$

If X,Y are monoids and we wish to preserve (1,1) as right and/or left identity we stipulate that

3a. $1^x = 1$ for all $x \in X$

and/or 3b. $y^1 = y$ for all $y \in Y$.

Now in order to interpret the semi-direct product of semi-groups in terms of machines we are forced to the configuration of Fig. 3 which appears as:



The reader familiar with (semi-) group theory may observe that the independencies involved in this configuration (the fact that Y does not affect M[X] and X does not directly affect M[Y]) are inherent in the notion of semi-direct product extensions though not true of extensions in general). Further, conditions 1 and 3a will satisfy the admissibility criterion for Z, and either of conditions 2 or 3b will assure that the hypothesis of Proposition 1 holds. Thus as expected, all semi-group semi-direct products can be interpreted as special cases of machine semi-direct products (preserving monoid-machinedness) but not conversely.[3] This means that it may be possible to obtain more efficient decompositions by using machine rather than semi-group semi-direct products. The Wreath product (also a semi-direct product) decomposition of groups is a case in point, as will be shown.

We now proceed to apply the preceeding ideas to the series-parallel decomposition of automata. Our approach is both a simplification and generalization of certain of the constructions employed in the Krohn-Rhodes theory of decomposition. In particular we generalize the notion of decomposition by subnormal series to monoids and show how to decompose group machines more efficiently than can be done using the Wreath product. This latter method was first discovered by R. Bayer (see [4] for both the Wreath product and Bayer's construction) but we show how it pops out of our more general approach.

Let K be a submonoid of a monoid X, with $\underset{K}{\equiv}$ the right congruence defined above. To each equivalence class [x] we assign a unique representative $x' \in [x]$ (called a leader in group theory), i.e., $x_1 \underset{K}{\equiv} x_2$ iff $x_1' = x_2'$. Now every $x \in X$ belongs to some $[x] \subseteq Kx$, hence has a representation $x = k_x x'$ for

[3] Because associativity $(y^{x_1 x_2} = (y^{x_2})^{x_1})$ is required of semi-group, but not machine products.

some $k_x \in K$. We shall assume for simplicity that only one such k_x exists. In the more general case an obvious but more complicated construction is involved which does not add anything new to the basic idea. We have the following easy but important lemmas:

Lemma 1. $(xy)' = (x'y)'$ (for all $x,y \in X$).

Proof. $x \underset{K}{\equiv} x'$ implies $xy \underset{K}{\equiv} x'y$ implies $(xy)' = (x'y)'$ Q.E.D.

Lemma 2. $k_{xy} = k_x k_{x'y}$ (for all $x,y \in K$).

Proof. $x = k_x x'$

$$\begin{aligned} \text{implies } xy &= k_x x'y \\ &= k_x k_{x'y}(x'y)' \\ &= k_x k_{x'y}(xy)' \text{ using Lemma 1.} \end{aligned}$$

Q.E.D.

We want to decompose monoid machines with monoid X. To do this it is convenient to consider machines of the form $M[X] = \langle X, X \rangle$. There is little lost in this, for with a little more care we can obtain the same results for machines whose input monoid is freely generated by the letters (i.e., the carrier) of X, which is the more standard formulation.

Theorem 1. Let X be a monoid and K a submonoid of X. There is an isomorphism from M[X] into a semi-direct product of M[X/K] and M[K].

Proof. That the code $x \leftrightarrow ([x],k_x)$ is a one-one representation for every $x \in X$ follows from the fact $x = k_x x'$. Now define the connecting map $Z : (X/\equiv_K, X) \to K$ to be $y^{[x]} = k_{x'y}$ for all $x,y \in X$. Checking that Z is admissible we have $1^{[x]} = k_{x'} = 1$ (since $x' = 1 \circ x'$) and $(y_1y_2)^{[x]} = k_{x'y_1y_2} = k_{x'y_1}k_{(xy_1)'y_2} = y_1{}^{[x]}y_2{}^{[xy_1]}$, using Lemmas 1 and 2. Now note that for all $x,y \in X$

$$\begin{aligned} ([x],k_x)y &= ([x]y, k_x y^{[x]}) \\ &= ([xy], k_x k_{x'y}) \\ &= ([xy], k_{xy}) \end{aligned}$$

which shows that every transition $x \circ y \to xy$ is correctly effected. Q.E.D.

We recall that M[X/K] is a monoid machine if K is a normal submonoid. Just as in the restricted case for groups, we may apply the above procedure to decompose both M[X/K] and M[K], thus obtaining a "subnormal" series of monoids and corresponding monoid machines. (Of course, a question of uniqueness, established for group decomposition, arises.) Decomposition by this method must halt when a "simple" monoid is reached. A monoid X is simple if for any normal submonoid K, either $X/\equiv_K \cong X$ or $X/\equiv_K \cong 1$.

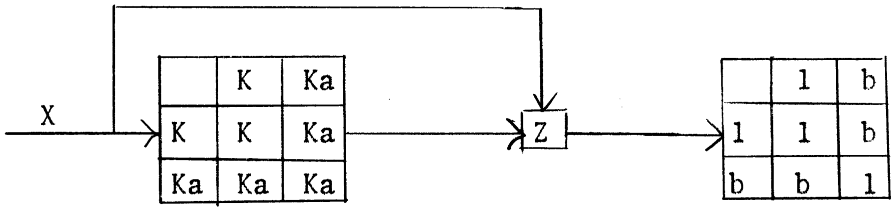
For finite automata, the Krohn-Rhodes theory shows that no new primitives are introduced in this way (assuming that the "simple" monoids are decomposed in the standard way). However, decomposition based on normal submonoids, where possible, is more efficient than that based on the existence of left ideals[4] in that it does not require the introduction or reset inputs. For infinite automata, where the existence of appropriate left ideals and sub-semigroups is not guaranteed, decomposition based on normal submonoids may provide an avenue where other procedures fail.

As an example, the monoid of Fig. 1 has classes K = {1,b}, Ka = {a,ba} such that Ka∘Ka = Ka. Choosing representatives of K and Ka as "1" and "a" respectively, we obtain the code between M[X] and $M[X/K] \otimes_Z M[K]$ as $1 \leftrightarrow (K,1)$, $b \leftrightarrow (K,b)$, $a \leftrightarrow (Ka,1)$, $ba \leftrightarrow (Ka,b)$. The Z map is given by

y	y^K	y^{Ka}
a	1	1
b	b	1

and the monoid machine decomposition is

[4]For a finite semigroup S, if S is neither cyclic nor left simple then there exists a proper left ideal $T \subset S$, $T \neq S$ and a proper sub-semigroup $V \subset S$, $V \neq S$ so that $S = T \cup V$. M(S) is then simulated by M(V) with added reset in series with $M(T \cup 1)$. See [5] or [6] for details.



We note that in the case of X being a group, k_x is directly computable as $k_x = x(x')^{-1}$, and our construction requires only the use of one copy of M[X/K] and one copy of M[K]. The Wreath product construction however also requires one copy M[X/K] but n copies of M[K] where $n = |X|/|K|$. The reason for this inefficiency, alluded to before, lies in the relation between semi-direct products of machines and those of monoids. The Wreath product decomposition on a group X, is a semi-direct product of X/K and $\pi_{i=1}^{n} K_i$, a direct product of n copies of K. Let $(v'_1, \ldots, v'_i, \ldots, v'_n)$ be an ordered set of representatives of the cosets $Kv'_1, \ldots, Kv'_i, \ldots, Kv'_n$. The code $x \leftrightarrow (Kx,(\ldots k_{v'_i x} \ldots))$ is a one-one map between X and $X/K \circledS_Z K$. The Z map is given by

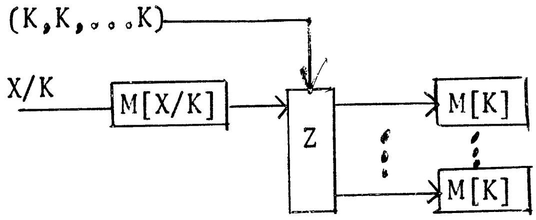
$$(k_{v'_1 x_1},\ldots,k_{v'_i x_1},\ldots,k_{v'_n x_1})^{Kx} = (k_{(vx_1)'x_1},\ldots,k_{(v_i x)'x_1},\ldots,k_{(v_n x)'x_1})$$

So that

$$\begin{aligned} xx_1 &\leftrightarrow (Kx,(\ldots k_{v'_i x}\ldots))\ (Kx_1,(\ldots k_{v'_i x_1}\ldots)) \\ &\leftrightarrow (KxKx_1,(\ldots k_{v'_i x}\ldots)(\ldots k_{v'_i x_1}\ldots)^{Kx}) \\ &\leftrightarrow (KxKx_1,(\ldots k_{v'_i x} k_{(v_i x)'x_1}\ldots) \\ &\leftrightarrow (Kxx_1,(\ldots k_{v'_i xx_1}\ldots)). \end{aligned}$$

(using Lemma 2)

as required. Implementing this as a machine decomposition yields the following configuration:



We see that the inefficiency is due to the fact input X is not allowed to pass to the Z map directly (as in the case in the construction of Theorem 1). To make up for this an expensive coding is required allowing Z to reconstruct the necessary information.[5] All this is due to the restrictions placed on semi-direct products of monoids which are not necessary for semi-direct products of machines.

In case K is normal submonoid of a monoid X but $\equiv_K$ coincides with the identity ($x,y \in X$ implies $x \equiv_K y$ iff $x = y$) it is still possible to obtain the coarser two sided congruence $\equiv_{\underline{K}}$ defined for all $x,y \in X$ by

$$x \equiv_{\underline{K}} y \text{ iff there are } k_1, k_2 \in K \text{ such that } k_1 x = k_2 y.$$

We leave it to the reader to verify that $\equiv_{\underline{K}}$ is indeed a congruence or see [2] and that $\equiv_K$ refines $\equiv_{\underline{K}}$. Using $\equiv_{\underline{K}}$ we can obtain a decomposition of M[X] which, however, requires feedback, a subject treated in a coming report. Roughly this is because every x is completely specified by two (rather than one) elements of K in addition to its representative and the next value of one of these can be computed only with knowledge of all three quantities. Because the construction is rather messy, we relegate its discussion to the Appendix.

[5] Note that Z satisfies the associativity requirement by inducing permutations on the vector $(k_{v_1 x}, \ldots, k_{v_n x})$ which contains all the necessary information to compute the transition.

APPENDIX

Let X be a monoid, K a normal submonoid of X and the congruence $\underset{K}{\equiv}$ as defined above. Let x' denote the unique representative of [x] in $X/\underset{K}{\equiv}$. Then every $x \in X$ satisfies at least one equation of the form

$$\ell_x x = k_x x'$$

for some $\ell_x, k_x \in K$.

Let $K(x) = \{(\ell_x, k_x) \mid \ell_x x = k_x x'\}$. We shall impose the condition that if x,y $(x \neq y)$ are in the same class, their associated sets K(x), K(y) are disjoint, viz.

*) $x \underset{K}{\equiv} y$ and $x \neq y$ implies $K(x) \cap K(y) = \emptyset$ for all $x,y \in X$.

<u>Lemma 3.</u> $(\ell_x, k_x) \in K(x)$ implies $(\ell_x \ell_{(xy)}, k_x k_{x'y}) \in K(xy)$ where $\ell_{(xy)}$ is some element in K depending on x and y, and $k_{x'y}$ is a right member of some $(\ell_{x'y}, k_{x'y}) \in K(x'y)$.

<u>Proof.</u> Since K is normal ($xK = Kx$ for all $x \in X$) the equation

$$\ell_{x'y} x'y = x'y\ell_0$$

has a solution $\ell_0 \in K$. By definition (of K(x'y))

$$\ell_{x'y} x'y = k_{x'y}(xy)'$$

(using Lemma 1). So

$$x'y\ell_0 = k_{x'y}(xy)'$$

implying

$$k_x x'y\ell_0 = k_x k_{x'y}(xy)'$$

implying

$$\ell_x xy\ell_0 = k_x k_{x'y}(xy)'$$

(using the definition of K(x)) implying

$$\ell_x \ell_{(xy)} xy = k_x k_{x'y}(xy)'$$

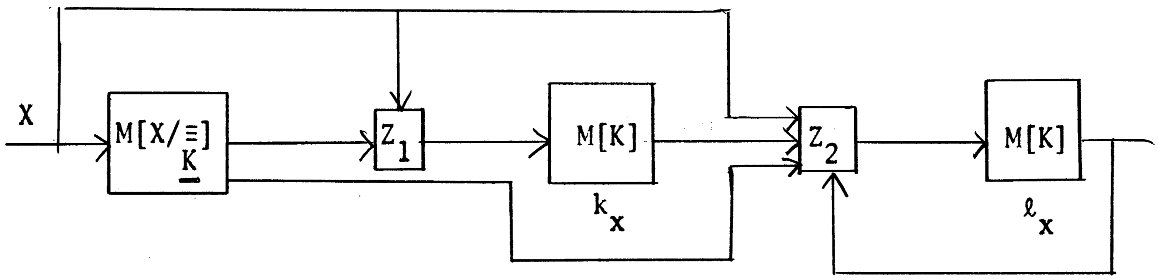
(again using the normality of K) implying

$$(\ell_x \ell_{(x,y)}, k_x k_{x'y}) \in K(xy).$$

Q.E.D.

Theorem 3. Let K be a normal submonoid of a monoid X such that condition *) holds. M[X] is a homomorphic image of a submachine of a semi-direct composition of $M[X/\underset{K}{\equiv}] = \langle X\ X/\underset{K}{\equiv}\rangle$, M[K], and $M^f[K]$ (where M^f denote a machine with feedback, see definition 1, Section 2).

Proof. The required composition is diagrammed as



The relevant submachine will have state set

$$Q = \{([x],\ell_x,k_x)\,|\,[x] \in X/\underset{K}{\equiv}, (\ell_x,k_x) \in K(x)\}.$$

The connecting map $Z_1 : (X, X/\underset{K}{\equiv}) \to K$ is defined by

$$y^{[x]} = k_{x'y}$$

for all $y \in X$, $[x] \in X/\underset{K}{\equiv}$ where $k_{x'y}$ is arbitrarily chosen but fixed right member of some $(\ell_{x'y},k_{x'y}) \in K(x'y)$. The map $Z_2 : (X, X/\underset{K}{\equiv}, K, K) \to K$ is defined by

$$y^{([x],\ell_x,k_x)} = \ell_{(xy)}$$

for all $y \in X$, $([x],k_x,\ell_x) \in Q$ where $\ell_{(xy)}$ may be computed as in Lemma 3, i.e., $(\ell_x\ell_{(xy)},k_xk_{x'y}) \in Kxy$. The operation of the machine will now be

$$\begin{aligned}([x],\ell_x,k_x)y &= ([x]y,\ell_x y^{([x],\ell_x,k_x)},k_x y^{[x]}) \\ &= ([xy],\ell_x\ell_{(xy)},k_xk_{x'y}) \\ &= ([xy],\ell_{xy},k_{xy})\end{aligned}$$

where $(\ell_{xy},k_{xy}) = (\ell_x\ell_{(xy)},k_xk_{x'y}) \in K(xy)$.

It is now obvious that the identification

$$([x],\ell_x,k_x) \to x$$

is a homomorphism onto M[X]. The condition *) guarantees that no ambiguity arises, i.e.,

$$x \neq y \text{ implies } ([x], k_x, \ell_x) \neq ([y], \ell_y, k_y).$$

Note that although feedback is required around the last M[K], no resets have been added to it so that its monoid has not been enlarged nor has it been made to act merely as a delay.

Finally, we observe that condition *) is equivalent to requiring certain left cancellation properties hold with respect to K.

Proposition:[1] *) holds if, and only if, for all $x, y \in X$, $k \in K$, $x \underset{K}{\equiv} y$ and $kx = ky$ implies $x = y$.

Proof. Suppose $x \neq y$, $x \underset{K}{\equiv} y$ and $(\ell, k) \in K(x) \cap K(y)$. Then $\ell x = kx'$ and $\ell x = kx$ so $\ell x = \ell y$ implies $x = y$ a contradiction.

Conversely, suppose $x \underset{K}{\equiv} y$, $x \neq y$, and $kx = ky$ for some $k \in K$. By normality, find some $\ell_0 \in K$ such that

$$kx = x\ell_0.$$

Then $\ell_x x = k_x x'$

implies $\ell_x x \ell_0 = k_x x' \ell_0$

implies $\ell_x k = k_x x' \ell_0$

implies $\ell_x kx = k_x x' \ell_0$.

(again using normality).

But also $\ell_x ky = k_x \ell_0' x'$

so $(\ell_x k, k_x \ell_0') \in K(x) \cap K(y)$ and *) does not hold. Q.E.D.

[1]The observation and proof of this proposition are due to Stewart Bainbridge.

REFERENCES

1. Y. Give'on, "Categories of Semimodules," Systems Research Center, Case Institute of Technology, SRC 100-C-66-40.

2. B. P. Zeigler, "Degenerate Automata: Some Relationships Involving Semigroup Order and Regular Events," University of Michigan Technical Report.

3. K. H. Hofmann, and P. S. Mostert, _Elements of Compact Semigroup_, C. E. Merrill Books, Inc., 1966.

4. M. Arbib, _The Wreath Product Made Simple_, Stanford University.

5. K. Krohn and J. Rhodes, "Algebra Theory of Machines: I. Prime Decomposition Theorem for Finite Semigroups and Machines," _Trans. Amer. Math. Soc._ _116_ (1965) 450-464.

6. M. Arbib, _The Theory of Abstract Automata_.

DISTRIBUTION LIST

(One copy unless otherwise noted)

Technical Library
Director Defense Res. & Eng.
Room 3C-128, The Pentagon
Washington, D.C. 20301

Defense Documentation Center 20
Cameron Station
Alexandria, Virginia 22314

Chief of Naval Research 2
Department of the Navy
Washington 25, D.C.
Attn: Code 437, Information Systems Branch

Director, Naval Research Laboratory 6
Technical Information Officer
Washington 25, D.C.
Attention: Code 2000

Commanding Officer 10
Office of Naval Research Branch Office
Box 39, Fleet Post Office
New York, New York 09510

Commanding Officer
ONR Branch Office
207 West 24th Street
New York 11, New York

Office of Naval Research Branch Office
495 Summer Street
Boston, Massachusetts 02110

Naval Ordnance Laboratory
White Oaks, Silver Spring 19
Maryland
Attn: Technical Library

David Taylor Model Basin
Washington,D.C. 20007
Attn: Code 042, Technical Library

Naval Electronics Laboratory
San Diego 52, California
Attn: Technical Library

Dr. Daniel Alpert, Director
Coordinated Science Laboratory
University of Illinois
Urbana, Illinois

Air Force Cambridge Research Labs
Laurence C. Hanscom Field
Bedford, Massachusetts
Attn: Research Library, CRMXL R

U. S. Naval Weapons Laboratory
Dahlgren, Virginia 22448
Attn: G. H. Gleissner, Code K4
Asst. Dir. for Computation

National Bureau of Standards
Data Processing Systems Division
Room 239, Building 10
Washington 25, D.C.
Attn: A. K. Smilow

George C. Francis
Computing Laboratory, BRL
Aberdeen Proving Ground, Maryland

Office of Naval Research
Branch Office, Chicago
230 North Michigan Avenue
Chicago, Illinois 60601

Commanding Officer
ONR Branch Office
1030 E. Green Street
Pasadena, California

Commanding Officer
ONR Branch Office
1076 Mission Street
San Francisco, California 94103

DISTRIBUTION LIST (Concluded)

The University of Michigan
Department of Philosophy
Attn: Professor A. W. Burks

National Physical Laboratory
Teddington, Middlesex, England
Attn: Dr. A. M. Uttley, Supt.
Autonomics Division

Commanding Officer
Harry Diamond Laboratories
Washington, D.C. 20438
Attn: Library

Commanding Officer and Director
U. S. Naval Training Device Center
Orlando, Florida 32813
Attn: Technical Library

Department of the Army
Office of the Chief of Research and Development
Pentagon, Room 3D442
Washington 25, D.C.
Attn: Mr. L. H. Geiger

National Security Agency
Fort George G. Meade, Maryland
Attn: Librarian, C-332

Lincoln Laboratory
Massachusetts Institute of Technology
Lexington 73, Massachusetts
Attn: Library

Office of Naval Research
Washington 25, D.C.
Attn: Code 432

Dr. Kenneth Krohn
Krohn Rhodes Research Institute, Inc.
328 Pennsylvania Avenue, S. E.
Washington 13, D. C.

Dr. Larry Fogel
Decision Science, Inc.
6508 Pacific Highway
San Diego, California

National Bureau of Standards
Applications Engineering Section
Washington 25, D. C.
Attn: Miss Mary E. Stevens

Unclassified

Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* Logic of Computers Group The University of Michigan Ann Arbor, Michigan 48104	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

DECOMPOSITIONS OF AUTOMATA USING NORMAL SUBMONOIDS

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

5. AUTHOR(S) *(Last name, first name, initial)*

Zeigler, Bernard

6. REPORT DATE December, 1967	7a. TOTAL NO. OF PAGES 14	7b. NO. OF REFS 6

8a. CONTRACT OR GRANT NO. Nonr 1224(21)	9a. ORIGINATOR'S REPORT NUMBER(S) 03105-50-T
b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
d.	

10. AVAILABILITY/LIMITATION NOTICES

Qualified requesters may obtain copies of this report from DDC.
Distribution of this document is unlimited

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Office of Naval Research Department of the Navy Washington 25, D.C.

13. ABSTRACT

Aspects of decomposition of automata using normal monoids are discussed. A simplified cascade connection procedure is given which can be specialized to the case of normal subgroups, but which introduces some novel features, e.g., the use of feedback in the tail machine, when applied to normal submonoids in general. Differences in efficiency between decompositions based on semidirect products of machines versus those of semigroups are also discussed.

DD FORM 1 JAN 64 1473

Unclassified

Security Classification

Unclassified

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
automata						
decompositions						
monoids						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2*a*. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2*b*. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7*a*. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7*b*. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8*a*. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8*b*, 8*c*, & 8*d*. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9*a*. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9*b*. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through ________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through ________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through ________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional.

Unclassified

Security Classification