STRUCTURE OF EVENTS AND AUTOMATA

Thesis for the Degree of Ph. D.
MICHIGAN STATE UNIVERSITY
BOBBY GENE REYNOLDS
1968

This is to certify that the

thesis entitled

STRUCTURE AND EVENTS AND AUTOMATA

presented by

Bobby Gene Reynolds

has been accepted towards fulfillment
of the requirements for

Ph.D. degree in E. E.

William Kilmer
Major professor

Date September 3, 1968

This is to certify that the

thesis entitled

STRUCTURE AND EVENTS AND AUTOMATA

presented by

Bobby Gene Reynolds

has been accepted towards fulfillment
of the requirements for

Ph.D. degree in E.E.

William Kilmer
Major professor

Date September 3, 1968

O-169

ABSTRACT

STRUCTURE OF EVENTS AND AUTOMATA

By

Bobby Gene Reynolds

The relations between certain structural aspects of automata and their accepted events are investigated.

A general repetitive machine is an automaton having a path from every final state to the start state. The general repetitive machines contain as a proper subclass the strongly connected machines. General repetitive events are defined in terms of certain "factorization" properties of the associated regular expressions, and a one-to-one onto correspondence is shown between general repetitive machines and general repetitive events.

Definite and ultimate-definite general repetitive machines are investigated, and a number of properties exhibited. All ultimate-definite automata are synchronizable; that is, there is an input sequence which causes the machine to move to a predetermined state regardless of the state of the machine before applying the sequence. Ultimate-definite general repetitive machines are shown to be strongly connected and hence synchronizable to the start state.

The structure question is also pursued from the viewpoint of a machine's cascade decomposition (Zeiger's method). An input to a component machine of a Zeiger cascade either permutes the states or resets to one state. It is shown that certain subsets of a machine that are permuted under some input appear as cover elements somewhere in a Zeiger cascade, and permutations of the states of component machines appear when these subsets are not singletons. From this, necessary and sufficient conditions are obtained for a machine to have a Zeiger cascade with no permutations other than those assigned arbitrarily. Finally, it is shown that an event R is definite if and only if no Zeiger cascade for the machine accepting R has permutations.

STRUCTURE OF EVENTS

AND AUTOMATA

By

Bobby Gene Reynolds

A THESIS

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Department of Electrical Engineering

1968

To Pat, my wife

ACKNOWLEDGEMENTS

I am pleased to acknowledge Dr. William Kilmer, chairman of my Guidance Committee, for his interest and helpful discussions during the course of my research.

My thanks also go to the following men for serving on the Guidance Committee: Dr. R.C. Dubes, Dr. J.S. Frame, Dr. H.E. Koenig, and Dr. J. Stapleton.

Finally, I wish to express my gratitude to my wife Pat for her patience and encouragement throughout this program.

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

The development of finite automata theory in recent years is a result of investigations of discrete-parameter systems; that is, systems whose input, output, and state variables each assume a finite number of values. The proliferation of digital devices and their usage in both military and commercial applications have provided motivation to study such devices on an abstract level, independent of any method of physical realization, in order to determine their capabilities and limitations.

The aim of this study is to examine the relations between certain structural aspects of automata and the events (sets of input sequences) accepted by them. In this chapter most of the notation is established, and basic concepts of automata theory are reviewed. In Chapter 2 a general repetitive machine is defined in terms of paths in the state graph of the machine, and properties of the event accepted by the machine are studied. The important class of strongly connected machines is a subclass of the general repetitive machines. Results from Chapter 2 are applied in Chapter 3 to definite and ultimate-definite automata. In Chapter 4 the cascade decomposition of automata is examined, and the relations between permutations of state subsets of an automaton and permutations of the states of component machines in the cascade are investigated. Finally, it is shown that the definite automata are precisely those whose cascade decompositions have no component machines whose states are permuted under some input.

1.1 Notation and Basic Concepts

Let A and B be sets. The set-theoretic union of A and B is written $A \cup B$ or $A + B$, and their intersection is $A \cap B$. $|A|$ denotes the number of elements in A.

An alphabet Σ is a finite non-empty set of symbols. The concatenation of symbols σ and τ is the sequence $\sigma\tau$. Σ^* is the set of all finite sequences obtained by concatenating symbols from Σ, including the null (or empty) sequence Λ, and for $B \subseteq \Sigma$, B^* is similarly defined. A sequence from Σ^* is a <u>tape</u>. The length of a sequence $x \in \Sigma^*$, $\ell(x)$, is the number of symbols in x. If x is a tape, then x^k denotes the tape $xx\ldots x$ (k times), and $x^0 = \Lambda$. If Σ^k denotes the set of all tapes in Σ^* of length k, then $\Sigma^* = \{\Lambda + \Sigma + \Sigma^2 + \Sigma^3 + \ldots\}$. Σ^* is thus a monoid under concatenation, with identity Λ which has the property $\Lambda x = x\Lambda = x$ for any $x \in \Sigma^*$. Any subset P of Σ^* is an <u>event</u>. The <u>product</u> or concatenation of events P and Q is $PQ = \{xy : x \in P, y \in Q\}$. If x is a tape, the event Px is similarly defined: $Px = \{wx : w \in P\}$. Thus we make no notational distinction between a tape x and the singleton set $\{x\}$.

Let $x = \sigma_1\ldots\sigma_m$ be a tape. For $1 \le j \le k \le m$ define ${}_jx_k$ by ${}_jx_k = \sigma_j\ldots\sigma_{k-1}$. Then ${}_jx_k$ is a <u>subtape</u> of x. Also ${}_1x_k$ is a <u>prefix</u> of x and ${}_jx_m$ is a <u>suffix</u> of x; they are <u>proper</u> iff $k \le m$ and $j > 1$ respectively. ("Iff" is an abbreviation for "if and only if".)

$A = (Q_A, M, s_0, F)$ is a <u>finite automaton</u>, and is defined over an input alphabet Σ. Q_A is a finite non-empty set of <u>states</u> of A. M is a map from $Q_A \times \Sigma$ into Q_A, and is called the <u>move</u> or <u>transition function</u>. If $s,t \in Q_A$ and $\sigma \in \Sigma$, $M(s,\sigma) = t$ means that if A is in state s and input σ is applied, A changes to state t. $s_0 \in Q_A$ is the <u>start state</u>, and $F \subseteq Q_A$ is the set of <u>final states</u>. M extends naturally to $Q_A \times \Sigma^*$ as follows: for each $x \in \Sigma^*$, $\sigma \in \Sigma$ and $s \in Q_A$, $M(s,x\sigma) = M(M(s,x),\sigma)$. For any state s, $M(s,\Lambda) = s$. For $B \subseteq Q_M$ and $x \in \Sigma^*$ we define $M(B,x) = \{M(s,x) : s \in B\}$. In Chapter 4 the letters M, K and L will be used to denote finite automata; the context will clearly show whether M denotes an automaton or a move function. We use the words <u>finite automaton</u>, <u>automaton</u>, <u>f.a.</u>, and <u>machine</u> interchangeably.

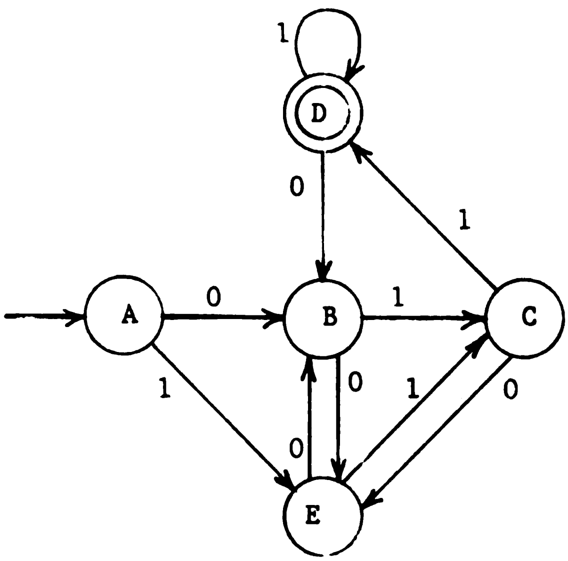
If the automaton is in state s_0 and tape $x = \sigma_1\ldots\sigma_m$ is applied to the input, A goes through the states s_0, $M(s_0,\sigma_1)$, $M(s_0,\sigma_1\sigma_2),\ldots,M(s_0,x)$. If $M(s_0,x) \in F$, the tape is <u>accepted by the automaton A</u>; otherwise it is <u>rejected</u>. A thus dichotomizes Σ^* into two events, the accepted tapes and the rejected tapes. An event which is the accepted set of some f.a. is a <u>regular</u> event. The automaton A whose accepted set is R is sometimes written $A(R)$. More generally, a tape x is accepted by the state s iff $M(s,x) \in F$.

The f.a. defined above has an output associated with each state; the output is "yes" or "1", say, if the automaton is in a state in F, and "no" or "0" otherwise. Such automata are called Moore machines, as distinguished from Mealy machines whose outputs are associated with the transitions between states. Moore machines are used exclusively in this study.

Automata can be represented by either a state graph or a flow table, as illustrated in Figure 1.1.1. We will use both representations, choosing the more convenient form for each situation. In a state graph, circles denote states and a final state is denoted by two concentric circles. Transitions between states are denoted by arrows, and each arrow is labelled with the symbol causing the transition. If $M(s,\sigma) = t$, an arrow is drawn originating at the circle representing state s and pointing to the circle representing state t, and σ is written near the arrow. The start state s_0 is distinguished by an arrow entering the circle for s_0 but not originating from any circle. A flow table has a column listing the states of Q_A and, for each input symbol, a column listing "next states" of A under that input. The final column designates final states by "1" and non-final states by "0".

A state t is reachable or accessible from state s iff $M(s,x) = t$ for some tape x. The set of states reachable from state s is $R(s) = \{M(s,x): x \in \Sigma^*\}$. A is connected iff $R(s_0) = Q_A$, and strongly connected iff $R(s) = Q_A$ for every state s. There is a path from s to t iff t is reachable from s. $B \subseteq Q_A$ is a strongly connected subset iff $B \subseteq R(s)$ for all $s \in B$.

Two states s and t of A are said to be indistinguishable or equivalent iff for all $x \in \Sigma^*$, $M(s,x) \in F$ iff $M(t,x) \in F$. Otherwise they are distinguishable, and there is a tape y such that $M(s,y) \in F$ and $M(t,y) \notin F$ or vice versa. There are several equivalent definitions of indistinguishability. In Figure 1.1.1 the states B and E are indistinguishable. These can be replaced with a single state B as in Figure 1.1.2. An automaton is reduced if it is connected and any two states are distinguishable. For every non-reduced

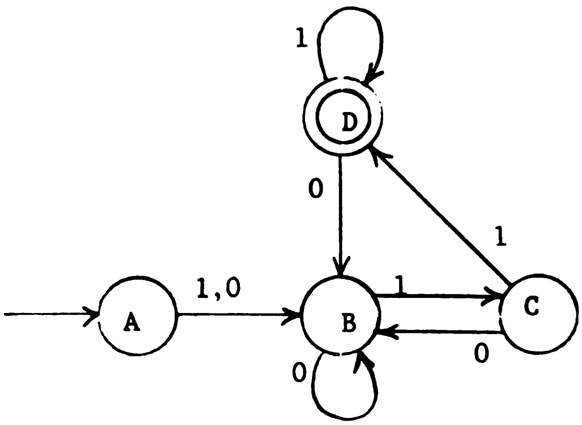


(a) State graph

	0	1	
A	B	E	0
B	E	C	0
C	E	D	0
D	B	D	1
E	B	C	0

(b) Flow table

Figure 1 1.1. State graph and flow table representations for an automaton with $\Sigma = \{0,1\}$



(a) State graph

	0	1	
A	B	B	0
B	B	C	0
C	B	D	0
D	B	D	1

(b) Flow table

Figure 1.1.2. Reduced version of the automaton in Figure 1.1.1.

automaton there is an equivalent reduced automaton which accepts exactly the same regular event. This study deals exclusively with reduced automata unless otherwise stated.

As stated earlier, the class of regular events consists of just those subsets of Σ^* which are accepted sets of finite automata. There are non-regular events, for example $\{0^n 1 0^n: n = 0,1,2,\ldots\}$ with $\Sigma = \{0,1\}$. Regular events are represented by <u>regular expressions</u>, which are constructed from an alphabet Σ by using only the union, dot (concatenation), and star operations a finite number of times. Thus, if $\sigma \in \Sigma$ then $\{\sigma\}$ is regular, and if P and R are regular, then $P + R$ (or $P \cup R$), $P \cdot R$ (or PR), and $P^* = \{\Lambda + P + P^2 + P^3 + \ldots\}$ are regular. No event is regular unless obtained in this fashion.

Finally, we review derivatives of events and state without proof several of their properties [2,3]. Let $P \subseteq \Sigma^*$ be an event. The <u>derivative of P</u> with respect to the tape $x \in \Sigma^*$, denoted $D_x(P)$, is defined by $D_x(P) = \{w: xw \in P\}$. For $\sigma, \tau \in \Sigma$, $D_\sigma(\sigma) = \Lambda$, and $D_\sigma(\tau) = \phi$ if $\sigma \neq \tau$; also, $D_\sigma(\Lambda) = \phi$, $D_\Lambda(\tau) = \tau$, and $D_\Lambda(P) = P$. If $\sigma \in \Sigma$ and $x \in \Sigma^*$, then $D_\sigma(D_x(P)) = D_{x\sigma}(P)$.

Define $\delta(P)$ by $\delta(P) = \phi$ if $\Lambda \notin P$ and $\delta(P) = \Lambda$ if $\Lambda \in P$. If Q is an event and $\sigma \in \Sigma$, the following properties of derivatives hold:

(1) $D_\sigma(P+Q) = D_\sigma(P) + D_\sigma(Q)$,

(2) $D_\sigma(P \cap Q) = D_\sigma(P) \cap D_\sigma(Q)$,

(3) $D_\sigma(PQ) = D_\sigma(P) \cdot Q + \delta(P) D_\sigma(Q)$,

(4) $D_\sigma(P^*) = D_\sigma(P) \cdot P^*$.

Let R be a regular event. A derivative of R is a regular event, and R has a finite number of derivatives. The number of derivatives of R is equal to the number of states in the reduced automaton $A(R) = (Q_A, M, s_0, F)$ accepting R. The relation between derivatives of R and states of A(R) is as follows. With the state $M(s_0, w) \in Q_A$ is associated the derivative $D_w(R)$. Then $D_w(R)$ is exactly the set of tapes accepted by the state $M(s_0, w)$, or the set of tapes which take A(R) from $M(s_0, w)$ to F. $D_w(R)$ is the <u>derivative</u>

corresponding to state $M(s_0,w)$; and $\Lambda \in D_w(R)$ iff $M(s_0,w) \in F$.

If $A(P) = (Q_A, M, s_0, F)$ is a f.a., not necessarily reduced, then two states $M(s_0,x)$ and $M(s_0,y)$ are indistinguishable iff $D_x(P) = D_y(P)$.

2. GENERAL REPETITIVE EVENTS AND MACHINES

Considerable research has been devoted to the class of strongly connected automata. One property of these machines is that the start state is accessible from every final state (since all states are accessible from every state). Since much of the "loop" structure of a strongly connected machine's state graph resides in this property, it would seem interesting to investigate all machines having the property. These are called <u>general repetitive machines</u>.

After precisely defining general repetitive machines we characterize the state graphs of those not strongly connected. Then we define a class of events called <u>general repetitive events</u>, and show that these are exactly the events accepted by general repetitive machines. Some properties of these events are exhibited and examples given. Next we characterize the machines in terms of derivatives of the events accepted by them, since many of the results in Chapter 3 are obtained by use of derivatives. Synchronization of automata is defined and discussed and, finally, closure properties of the class of general repetitive events are considered.

2.1. <u>State Graph Structure of General Repetitive Machines</u>

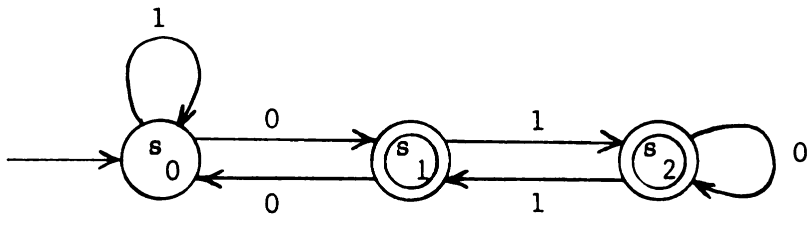
<u>Definition 2.1.1</u>: An automaton $A = (Q_A, M, s_0, F)$ is a <u>general repetitive machine</u> (GRM) iff for each state $s_i \in F$ there is a tape x_i such that $M(s_i, x_i) = s_0$.

In some instances a single tape will suffice to return a general repetitive machine from any final state to the start state; in others, different final states may require different tapes. These conditions are distinguished by the following definition.

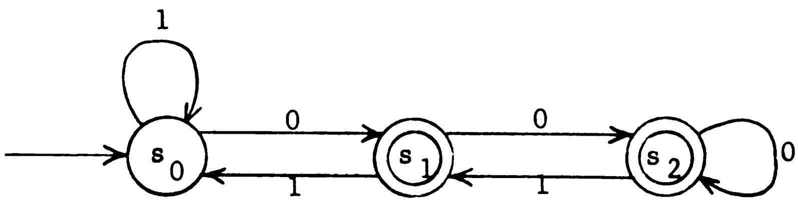
<u>Definition 2.1.2</u>: An automaton $A = (Q_A, M, s_0, F)$ is a <u>repetitive machine</u> (RM) iff for some tape x and for all $s \in F$, $M(s,x) = s_0$.

Both automata of Figure 2.1.1 are GRMs. The automaton of Figure 2.1.1(b) is an RM, since $M(F, 11) = s_0$, but that of Figure 2.1.1(a) is not; the RMs thus constitute a proper subclass of the GRMs.

The class of strongly connected machines is a proper subclass of the GRMs. We partition the class of GRMs into two subclasses, strongly



(a) GRM



(b) RM

Figure 2.1.1 . Examples of general repetitive machines.

connected and non-strongly connected, and designate them as Type I and Type II, respectively.

<u>Lemma 2.1.1</u>: Let $A(R) = (Q_A, M, s_0, F)$ be a GRM. If $A(R)$ has a final dead state, then $A(R)$ is a 1-state automaton. If $A(R)$ has a non-final dead state s_D, then $Q_A - \{s_D\}$ is a strongly connected subset of states.

<u>Proof</u>: Let s_D be a final dead state. Since $A(R)$ is a GRM, $s_0 \in R(s_D)$; hence $s_D = s_0$. But s_D is dead so $Q_A = \{s_0\}$.

Suppose s_D is a non-final dead state. Let $B = Q_A - \{s_D\}$. If $s \in B$, s is not dead, and for some $s_F \in F$, $s_F \in R(s)$. But

$s_0 \in R(s_F)$ and $B \subseteq R(s_0)$; hence $B \subseteq R(s)$. That is, every state in B is accessible from every state s in B, and so B is a strongly connected subset of Q_A.

Q.E.D.

Theorem 2.1.1: Every Type II GRM $A = (Q_A, M, s_0, F)$ consists of a single non-final dead state s_D and a strongly connected subset $Q_A - \{s_D\}$.

Proof: If A has no non-final dead state, then from every state s there is a path to some $s_F \in F$. There is also a path from every state in F to s_0, and from s_0 to every other state. Hence $R(s) = Q_A$ and A is strongly connected, contrary to the assumption that A is Type II. Thus A has a non-final dead state s_D, and only one since A is reduced. The rest follows from Lemma 2.1.1.

Q.E.D.

This theorem shows that the Type II GRMs are quite similar to the strongly connected machines, since all states but one lie in a strongly connected subset of Q_A.

2.2 General Repetitive Events

Definition 2.2.1: A repetitive event (RE) is a regular event P with the properties

(1) for some $x \in \Sigma^*$, $P = P(xP)^*$, and

(2) for any events R and Q, if $R \subseteq P$ and $RxQ \subseteq P$ then $Q \subseteq P$.

The tape x is called a return tape. The class of REs is denoted by $\hat{R}$.

Definition 2.2.2: A general repetitive event (GRE) is a regular event R with the properties

(1) $R = \bigcup_{i=1}^{m} P_i$, where each P_i is an RE with return tape x_i,

(2) for $1 \le i,j \le m$, P_i can be written $P_i = (P_j x_j)^* P_i$, and

(3) for $1 \le i,j \le m$ and for any events R_i and Q, if $R_i \subseteq P_i$ and $R_i x_i Q \subseteq P_j$ then $Q \subseteq P_j$.

The class of GREs is denoted by $\hat{G}$.

Observe that an RE is a GRE, and hence $\hat{R}$ is a proper subclass of $\hat{G}$. Also, no finite event belongs to either class.

A lemma is needed for later use.

Lemma 2.2.1: If P is an event and x is a tape, the following are equivalent:

(1) $P = P(xP)^*$

(2) $P = (Px)^*P$

(3) $PxP \subseteq P$.

Proof: The equivalence of 1 and 2 is seen as follows:

$$\begin{aligned} P(xP)^* &= P[\Lambda + xP + xPxP + \ldots] \\ &= P + PxP + PxPxP + \ldots \\ &= [\Lambda + Px + PxPx + \ldots]P \\ &= (Px)^*P. \end{aligned}$$

That 1 implies 3 is trivial. Conversely, if $PxP \subseteq P$, then $(PxP)xP \subseteq PxP \subseteq P$, or $P(xP)^2 \subseteq P$, and in like fashion we observe that $P(xP)^n \subseteq P$ for $n = 0,1,2,\ldots$, where $(xP)^0 = \Lambda$. Hence $P(xP)^* \subseteq P$. Obviously $P \subseteq P(xP)^*$, so $P = P(xP)^*$.

Q.E.D.

The following theorem relates GREs to GRMs.

Theorem 2.2.1: Let $A(R) = (Q_A, M, s_0, F)$ be an automaton. Then $A(R)$ is a GRM if and only if R is a GRE.

Proof: Let $F = \{s_{F_i}\}_{i=1}^m$ be the final states of $A(R)$.

Suppose $A(R)$ is a GRM. For $1 \le i \le m$ define P_i by $P_i = \{y \in \Sigma^*: M(s_0,y) = s_{F_i}\}$; that is, P_i would be the accepted set of A is s_{F_i} were the only final state. Then obviously $R = \bigcup_{i=1}^m P_i$. By Definition 2.1.1 there are tapes x_i for $1 \le i \le m$ such that $M(s_{F_i},x_i) = s_0$. We first show that each P_i is an RE. Suppose $w,y \in P_i$. Then $M(s_0,wx_iy) = M(s_{F_i},x_iy) = M(s_0,y) = s_{F_i}$. Hence for all $w,y \in P_i$, $wx_iy \in P_i$, which means $P_ix_iP_i \subseteq P_i$. By Lemma 2.1.1,

$P_i = P_i(x_iP_i)^*$. Now suppose R_i and Q are events such that $R_i \subseteq P_i$ and $R_ix_iQ \subseteq P_i$, and let $w \in R_i$ and $y \in Q$. Since $M(s_0,w) = s_{F_i}$ and $wx_iy \in P_i$, $s_{F_i} = M(s_0,wx_iy) = M(s_{F_i},x_iy) = M(s_0,y)$; hence $y \in P_i$. This shows that $Q \subseteq P_i$, and completes the proof that P_i is a repetitive event.

To show that $P_i = (P_jx_j)^*P_i$ for $1 \le j \le m$, let $y \in P_j$. Then $M(s_0,yx_j) = M(M(s_0,y),x_j) = M(s_{F_j},x_j) = s_0$. Hence for $z \in P_i$, $M(s_0,yx_jz) = M(s_0,z) = s_{F_i}$, and thus $P_jx_jP_i \subseteq P_i$. By a similar argument $(P_jx_j)^nP_i \subseteq P_i$ for $n = 0,1,2,\ldots,$ yielding $P_i = (P_jx_j)^*P_i$.

Finally, suppose R_i and Q are events such that $R_i \subseteq P_i$ and $R_ix_iQ \subseteq P_j$ for $1 \le i,j \le m$. Let $w \in R_i$ and $y \in Q$. Since $M(s_0,w) = s_{F_i}$ and $wx_iy \in P_j$, $s_{F_j} = M(s_0,wx_iy) = M(s_{F_i},x_iy) = M(s_0,y)$; hence $y \in P_j$, so that $Q \subseteq P_j$.

This completes the proof that R is a GRE.

Conversely, suppose R is a GRE which we can write as $R = \bigcup_{i=1}^{m} P_i$ according to Definition 2.2.2. Each P_i cannot necessarily be associated with a distinct final state in this case. Define instead the subsets F_i of F by $F_i = \{s_F \in F\colon M(s_0,y) = s_F$ for some $y \in P_i\}$. We also define $S_i = \{s \in Q_A\colon M(s_F,x_i) = s$ for some $s_F \in F_i\}$, where x_i is the return tape for P_i. The method of proof is to show that each state in S_i, $1 \le i \le m$, is indistinguishable from s_0 (that is, $S_i = \{s_0\}$ since $A(R)$ is reduced), so that x_i takes every state in F_i to s_0. We must show that for every s in S_i and y in Σ^*, and for $1 \le i \le m$, $M(s_0,y) \in F$ iff $M(s,y) \in F$.

Suppose $M(s_0,y) \in F$ for $y \in \Sigma^*$ and let s be any state in S_i. There is a tape $z \in P_i$ such that $M(s_0,zx_i) = s$, where x_i is a return tape for P_i. Since $y \in R$, $y \in P_j$ for some j. Hence $zx_iy \in P_ix_iP_j$, and since by definition $P_ix_iP_j \subseteq P_j$ we have $zx_iy \in P_j$. Thus $M(s_0,zx_iy) = M(M(s_0,zx_i),y) = M(s,y) \in F_j \subseteq F$. Hence $M(s_0,y) \in F$ implies $M(s,y) \in F$.

If $s \in S_i$ and $M(s,y) \in F$, choose $z \in P_i$ so that $M(s_0,zx_i) = s$. Then $M(s_0,zx_iy) = M(s,y) \in F$, so $zx_iy \in P_j$ for some j. Then $\{z\} \subseteq P_i$ and $\{z\}x_i\{y\} \subseteq P_j$, and by condition 3 of Definition 2.2.2 $\{y\} \subseteq P_j$. That is, $y \in P_j$, and hence $M(s_0,y) \in F$. Thus

$M(s,y) \in F$ implies $M(s_0,y) \in F$, completing the proof that s_0 and $s \in S_i$ are indistinguishable.

Hence $S_i = \{s_0\}$ for $1 \le i \le m$, and $M(F_i,x_i) = s_0$, showing that $A(R)$ is a GRM.

Q.E.D.

The above proof shows that the event R accepted by a GRM $A(R)$ can be expressed in terms of events P_i associated with individual final states s_{F_i}. That is, P_i is exactly the set of tapes taking $A(R)$ from s_0 to s_{F_i}, and x_i takes $A(R)$ from s_{F_i} to s_0. But a general repetitive event might be expressible in a number of ways satisfying the conditions of Definition 2.2.2, and some forms may not allow the component repetitive events to be associated with individual final states. One is assured, however, that a GRE form does exist so that each P_i is associated with a single final state.

It also follows that $A(R)$ is a repetitive machine iff R is a repetitive event. For if $A(R)$ is an RM we can select the same return tape x for each state, and straightforward manipulations of R show that it is an RE. The converse is easy.

We observe that many events P can be written $P(xP)^*$ for some $x \in \Sigma^*$ and yet do not satisfy condition 2 of Definition 2.2.1. In particular, any ultimate-definite event P (to be investigated in the next chapter) has the property $P = \Sigma^* P$, and for any tape x, $P = \Sigma^* P = \Sigma^* P(x\Sigma^* P)^* = P(xP)^*$. However, not all ultimate-definite events are REs. The property $P = P(xP)^*$ means that the state $s = M(s_F,x)$, where $s_F \in F$, accepts the event P. Condition 2 of the definition insures that s accepts exactly P. Ultimate-definite machines $A(P)$ will be seen to have the property that every state accepts P but only s_0 accepts exactly P.

Figure 2.2.1 shows an example of a repetitive machine $A(P)$. The repetitive event P is $P = 11(011)^*$. Letting $x = 0$, P can be written $P = P(xP)^* = 11(011)^*[011(011)^*]^*$. Condition 2 of Definition 2.2.1 is also satisfied since $A(P)$ is an RM.

Before presenting some properties of repetitive events we extend the definition of derivatives of events to the following.

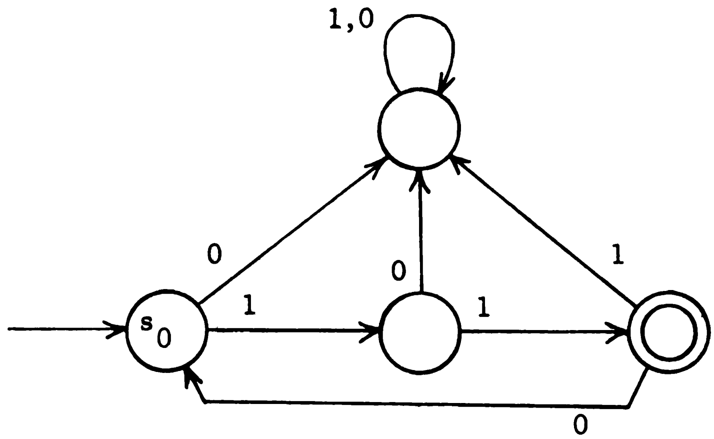


Figure 2.2.1. A repetitive machine $A(P)$ where $P = 11(011)^*$

Definition 2.2.3: Let W, Z be events. The derivative of W with respect to Z is $D_Z(W) = \{y \in \Sigma^*: zy \in W \text{ for some } z \in Z\}$. It is clear that $D_Z(W) = \bigcup_{z \in Z} D_z(W)$.

Conditions 1 and 2 of Definition 2.2.1 for a repetitive event P can be expressed in terms of the derivative $D_P(P)$; that is the subject of the next two theorems, which provide alternative ways of determining if a regular event is repetitive.

Theorem 2.2.2: Let P be a regular event. If $P = P(xP)^*$ for some tape x then $D_P(P) = D_P(P) \cdot (xP)^*$.

Proof: Suppose $P = P(xP)^*$ and let $y \in D_P(P) \cdot (xP)^*$. Then $y = uv$, where $u \in D_P(P)$ and $v \in (xP)^*$. By Definition 2.2.3 there is a tape $w \in P$ such that $wu \in P$; hence $wy = wuv \in P(xP)^* = P$, and so $y \in D_w(P) \subseteq D_P(P)$. Thus $D_P(P) \supseteq D_P(P) \cdot (xP)^*$. The reverse inclusion is obvious, proving $D_P(P) = D_P(P) \cdot (xP)^*$.

Q.E.D.

This result only provides a means, by contraposition, of determining that P is not repetitive. However, the condition $P = P(xP)^*$

for REs is the easier of the two conditions to verify. Given that $P = P(xP)^*$, the next theorem gives several necessary and sufficient conditions for condition 2.

<u>Theorem 2.2.3</u>: Let P be an event with the property $P = P(xP)^*$ for some tape $x \in \Sigma^*$. The following are equivalent, where R and Q are events and $y \in \Sigma^*$:

(1) $R \subseteq P$ and $RxQ \subseteq P$ imply $Q \subseteq P$,

(2) x is not a prefix of any tape in $D_P(P) - xP$,

(3) $P = D_{Px}(P)$,

(4) $xy \in D_P(P)$ iff $y \in P$.

<u>Proof</u>: (1) $\Rightarrow$ (2): Suppose $R \subseteq P$ and $RxQ \subseteq P$ imply $Q \subseteq P$. To show that x is not a prefix of any tape in $D_P(P) - xP$ we must show for each tape w that if $xw \in D_P(P)$ then $w \in P$.

If $xw \in D_P(P)$, by Definition 2.2.3 there is a tape $y \in P$ such that $yxw \in P$. That is, $\{y\} \subseteq P$ and $\{y\}x\{w\} \subseteq P$, so by hypothesis $\{w\} \subseteq P$. Hence $w \in P$.

(2) $\Rightarrow$ (3): Suppose $xw \in D_P(P)$ implies $w \in P$, and let $y \in D_{Px}(P)$. Now $D_{Px}(P) = \{y \in \Sigma^*: zxy \in P \text{ for some } z \in P\}$. Hence there is a tape $z \in P$ such that $zxy \in P$, and therefore $xy \in D_P(P)$. By hypothesis $y \in P$, showing $D_{Px}(P) \subseteq P$.

Since $P = P(xP)^*$ we have $PxP \subseteq P$. It follows that $P \subseteq D_{Px}(PxP) \subseteq D_{Px}(P)$. Hence $P = D_{Px}(P)$.

(3) $\Rightarrow$ (4): Suppose $P = D_{Px}(P)$.

Let $xy \in D_P(P)$. Then $y \in D_{Px}(P)$, so $y \in P$.

Conversely, let $y \in P$. Then $y \in D_{Px}(P)$, and hence for some tape $z \in P$ we have $zxy \in P$. Thus $xy \in D_P(P)$.

(4) $\Rightarrow$ (1): Suppose $xy \in D_P(P)$ iff $y \in P$. We assume R and Q are events such that $R \subseteq P$ and $RxQ \subseteq P$, and show $Q \subseteq P$. Using the fact that $D_R(RxQ) = \bigcup_{z \in R} D_z(RxQ)$, observe that $xQ \subseteq D_R(RxQ) \subseteq D_P(RxQ) \subseteq D_P(P)$. Now if $y \in Q$, then $xy \in D_P(P)$, and by hypothesis $y \in P$. Hence $Q \subseteq P$.

Q.E.D.

An interpretation of the event $D_P(P)$ may be of interest. If $A(P) = (Q_A, M, s_0, F)$ is the machine accepting P, then P is the event accepted by s_0. If $y \in P$, then $D_y(P)$ is the event accepted by the final state $M(s_0, y)$. Hence $D_P(P)$ is the union of the events accepted by the final states of $A(P)$; i.e., $D_P(P) = \{w \in \Sigma^*: M(s_F, w) \in F$ for some $s_F \in F\}$.

To close this section we show that events of a certain form are repetitive.

Theorem 2.2.4: Let $z, y \in \Sigma^*$. If z is a proper suffix of y, the events $P = zy^*$ and $R = y^*z$ are REs.

Proof: We can write $y = wz$, where $w \neq \Lambda$. Then $P = z(wz)^*$. Condition 1 of Definition 2.2.1 is verified by observing that

$$\begin{aligned} P(wP)^* &= z(wz)^*[w(z(wz)^*)]^* \\ &= z\{(wz)^*[wz(wz)^*]^*\} \\ &= z(wz)^* \\ &= P. \end{aligned}$$

To show condition 2, suppose $R \subseteq P$ and $RwQ \subseteq P$. Let $v \in R$ and $u \in Q$. Then $vwu \in P$, and hence $vwu = z(wz)^i$ for some i (possibly zero). Similarly $v \in P$ so $v = z(wz)^j$ for some j (possibly zero), and clearly $j < i$. Then $vwu = z(wz)^j wu = z(wz)^i$, and so $u = z(wz)^{i-j-1}$. Hence $u \in P$, proving $Q \subseteq P$.

It is similarly demonstrated that R is an RE.

Q.E.D.

2.3. Derivative Characterization of GREs and GRMs

Some of the structural properties of general repetitive events were exhibited in the last section. In this section we provide a more useful characterization in terms of derivatives so that these results can more easily be applied to ultimate-definite events in Chapter 3. Derivatives of regular events can be computed in a straightforward manner using the properties discussed in Section 1.1.

We present two results, one for GRMs and one for strongly connected automata.

Theorem 2.3.1: An automaton $A(R) = (Q_A, M, s_0, F)$ is a GRM iff for each $y \in R$, there is a tape $x \in \Sigma^*$ such that $D_{yx}(R) = R$.

Proof: Let $A(R)$ be a GRM. If $y \in R$ then $M(s_0, y) \in F$, and by Definition 2.1.1 there is a tape x such that $M(M(s_0,y),x) = s_0$. That is, $M(s_0, yx) = s_0 = M(s_0, \Lambda)$; hence $D_{yx}(R) = D_\Lambda(R) = R$.

Conversely, suppose that for each $y \in R$, there is a tape $x \in \Sigma^*$ such that $D_{yx}(R) = R$. For any final state s there is a tape $y \in R$ such that $M(s_0, y) = s$. Then for some tape x, $D_{yx}(R) = R = D_\Lambda(R)$, which means that $M(s_0, yx) = M(s_0, \Lambda) = s_0$. But $M(s_0, yx) = M(M(s_0,y),x) = M(s,x) = s_0$ and thus there is a path from s to s_0.

Q.E.D.

Of course one need not take derivatives of R with respect to every tape in R to apply the above theorem. There are finitely many distinct derivatives although R is infinite. This is clear from the correspondence between distinct derivatives of R and distinct states of $A(R)$. A straightforward procedure for testing an event R for the GRE property is as follows. First compute all derivatives of R according to the procedure given by Brzozowski [2]. For each derivative D of R which contains Λ compute all derivatives of D (again there are finitely many). If every such derivative D of R has a derivative equal to R, then $A(R)$ is a GRM.

The next theorem shows that one can check for strong connectedness by looking for empty derivatives of R.

Theorem 2.3.2: An automaton $A(R) = (Q_A, M, s_0, F)$ is strongly connected iff $A(R)$ is a GRM and all derivatives of R are non-empty.

Proof: A strongly connected automaton is obviously a GRM. Furthermore, a derivative $D_y(R)$ is exactly the set of tapes accepted by the state $M(s_0, y)$, and this set is non-empty since there is a path from $M(s_0,y)$ to F.

Conversely, suppose $A(R)$ is a GRM and $D_y(R) \neq \emptyset \; \forall y \in \Sigma^*$. Let s be any state, and let y be such that $M(s_0, y) = s$. For any tape $z \in D_y(R)$, $M(s,z) = s_F \in F$. Hence s_F is accessible from s, and since $A(R)$ is a GRM s_0 is accessible from s_F. Thus $s_0 \in R(s)$. But all states are accessible from s_0, and hence all states are accessible from s. That is, $R(s) = Q_A$, and $A(R)$ is strongly connected.

Q.E.D.

2.4. Synchronization of Finite Automata

The concept of automaton synchronization was motivated by the problem of driving a f.a. from an arbitrary unknown state to a known state by applying a predetermined input sequence (e.g. see [13]).

For example, if the present state of the automaton of Figure 2.1.1(a) is not known, no input sequence exists which will bring the automaton to a known state, since each input symbol permutes the state set. However, the input sequence 11 assuredly leaves the Figure 2.1.1(b) automaton in state s_0, and all subsequent states are known since each is reached from s_0 by following paths corresponding to the input symbols presented.

We introduce the concept at this point because it is intimately related to ultimate-definite GRMs, the subject of Chapter 3.

Definition 2.4.1: The tape (sequence) x is a synchronizing tape (sequence) for the f.a. $A = (Q_A, M, s_0, F)$ iff for each state $s \in Q_A$, $M(s,x) = M(s_0,x)$. In such a case, x is said to synchronize A.

Definition 2.4.2: The f.a. A is synchronizable iff there is a tape x which synchronizes A.

Thus, a synchronizing tape x for the f.a. A takes A from any state to $M(s_0,x)$.

The Figure 2.1.1 examples show that some, but not all, finite automata are synchronizable. A synchronizable automaton has infinitely many synchronizing tapes, for if x synchronizes A, so does wxy for all tapes w and y in Σ^*.

Liu [11] investigated some aspects of automaton synchronization, and established the following necessary and sufficient condition for synchronizability.

Theorem 2.4.1: $A = (Q_A, M, s_0, F)$ is synchronizable iff for every pair of states s and t in Q_A there is a tape x such that $M(s,x) = M(t,x)$. The tape x is said to merge states s and t.

The next result of Liu offers a second characterization of synchronizing tapes.

Theorem 2.4.2: The tape x synchronizes $A = (Q_A, M, s_0, F)$ iff $M(Q_A, x) = M(s_0, x)$. That is, $M(Q_A, x)$ must be a singleton subset of Q_A.

The proof is trivial.

We close this section with a result on synchronizability of Type II GRMs.

Theorem 2.4.3: Every Type II GRM $A = (Q_A, M, s_0, F)$ is synchronizable.

Proof: By Theorem 2.1.1 A has a dead state s_D, and $Q_A - \{s_D\}$ is strongly connected. Hence s_D is accessible from every state, since $s_D \in R(s_0)$ and $s_0 \in R(s)$ for all $s \in (Q_A - \{s_D\})$. Then for each state $s_i \in Q_A$, there is a tape x_i such that $M(s_i, x_i) = s_D$, where $M(s_D, \Lambda) = s_D$. If $s_i, s_j \in Q_A$ (not excluding $s_i = s_D$), and $M(s_j, x_i) = s_k$, then $M(s_i, x_i x_k) = s_D = M(s_k, x_k) = M(M(s_j, x_i), x_k) = M(s_j, x_i x_k)$, so that the tape $x_i x_k$ merges s_i and s_j. By Theorem 2.4.1 A is synchronizable.

Q.E.D.

2.5. Closure Properties of $\hat{R}$ and $\hat{G}$

Closure of the classes of repetitive and general repetitive events under the standard operations on events is important because closure of a class under an operation provides a means of forming new events of that class. Perhaps unfortunately, $\hat{R}$ and $\hat{G}$ are not closed under any of the usual event operations. This is stated in the next theorem. The proof will be delayed until Chapter 3 when more tools are developed to facilitate construction of counterexamples.

Theorem 2.5.1: Neither $\hat{R}$ nor $\hat{G}$ is closed under union, dot (concatenation), intersection, complementation or star.

3. APPLICATIONS TO DEFINITE AND ULTIMATE-DEFINITE AUTOMATA

Events which can be expressed in the form $R = P + \Sigma^* Q$ (P and Q finite) are called definite events, and have been investigated by several authors [1, 8, 15]. If $Q = \phi$, $R = P$ is initial definite; if $P = \phi$ and $Q \neq \phi$, $R = \Sigma^* Q$ is non-initial definite; and if $P \neq \phi \neq Q$, R is composite definite. Each finite event is initial definite.

Paz and Peleg [14] generalized the notion of non-initial definite events by allowing Q to be an infinite set, calling the resulting class of events ultimate-definite (u.d.).

In this chapter we investigate those definite and ultimate-definite events which are GREs. The chapter proceeds as follows. Preliminary material is presented in the first section, and in Section 3.2 the relationships among general repetitiveness, strong connectedness, synchronizability, and synchronizability to the start state are explored for definite and ultimate-definite automata. These results, most of which are for u.d. automata, are the main results of the chapter. The next section deals with structure properties of u.d. events and provides short cuts for constructing canonical representations of the union and intersection of u.d. events. Section 3.4 discusses closure properties of certain classes of events.

3.1. Basic Properties of Definite and Ultimate-Definite Events and Automata

This section contains some definitions and basic results concerning definite and u.d. events and their automata. All results except Theorem 3.1.1 are taken from Paz and Peleg [14] and Brzozowski [1]; they are listed as properties and no proofs are given here.

The class of u.d. events is denoted by U.D.

Property 1. $P \in$ U.D. iff $P = \Sigma^* P$.

Property 2. If P and Q are u.d., then $P + Q$ and $P \cap Q$ are u.d.

Property 3. If P is u.d. and Q is any event, then PQ is u.d.

Property 4. If P is u.d., then $P^* = \Lambda + P$.

Property 5. If P is u.d., then $P = \Sigma^*$ iff $\Lambda \in P$.

Definition 3.1.1: If P is u.d., then a representation $P = \Sigma^* Q$ of P is canonical iff no tape in Q is a proper suffix of a tape in Q.

If P is u.d., define the following subsets of P, where $\Sigma^0 = \{\Lambda\}$: $P_0 = P \cap \Sigma^0$; $P_{j+1} = P \cap \Sigma^{j+1} - \bigcup_{k=0}^{j} \Sigma^{j+1-k} P_k$, $j = 0,1,2,\ldots$

Property 6. An ultimate-definite event P has a unique canonical representation, denoted by $P = \Sigma^* P^t$, where $P^t = \bigcup_{j=0}^{\infty} P_j$.

The expression $\Sigma^*(0+11)$, where $\Sigma = \{0,1\}$, is in canonical form, but $\Sigma^*(0+10)$ is not since 0 is a proper suffix of 10. The canonical representation of $\Sigma^*(0+10)$ is $\Sigma^* 0$.

Definition 3.1.2: The set of non-null suffixes of a tape y will be denoted by S_y.

The following property will be used extensively in this chapter.

Property 7. Let P be u.d. with canonical representation $P = \Sigma^* P^t$. If $x \in \Sigma^*$, then $D_x(P) = P + \bigcup_{w \in S_x} D_w(P^t)$ and $P \cap \left[\bigcup_{w \in S_x} D_w(P^t) \right] = \emptyset$.

The preceding properties apply to the general class of u.d. events, which includes both regular and non-regular events. Only regular u.d. events will be considered hereafter.

Property 8. If $P = \Sigma^* P^t$, then P is regular iff P^t is regular.

Definition 3.1.3: An automaton $A(R) = (Q_A, M, s_0, F)$ is ultimate-definite iff R is u.d. $A(R)$ is definite iff R is definite.

Property 9. A reduced f.a. $A = (Q_A, M, s_0, F)$ is u.d. iff all accepted tapes x with length $\ell(x) \le |Q_A| (|Q_A| - 1)$ lead to F from any state.

Every definite event $R = P + \Sigma^* Q$ (i.e., P and Q are finite events) has a unique canonical representation obtained as follows [1]:

(1) Remove from P all tapes having suffixes in Q.

(2) Remove from Q all tapes having proper suffixes in Q.

(3) If $x \in P$ and $\Sigma^* x \subseteq \Sigma^* Q$, place x in Q and remove from Q all tapes having x as a proper suffix.

(4) Delete any redundant listing of tapes in P and Q.

Brzozowski [1] also assumed that the tapes in P and Q were ordered in their listing, but this is not an important restriction and we will not require it here.

The above procedure also yields a unique canonical representation if P or Q or both are infinite regular events. Furthermore, if $R = P + \Sigma^* Q$ is a canonical representation of R, then $\Sigma^* Q$ is in the canonical form for ultimate-definite events. We will consider all events $P + \Sigma^* Q$, where $P \neq \phi$, to be regular and in canonical form unless otherwise stated, although no special notation will be used.

Property 10. If $R = P + \Sigma^* Q$ is definite and in canonical form, then $P \cap \Sigma^* Q = \phi$.

Property 11. If $y \in \Sigma^*$ and $R = P + \Sigma^* Q$ is definite, the derivative $D_y(R)$ is definite.

The following theorem is the only new result included in this section.

<u>Theorem 3.1.1</u>: If P is a (possible non-regular) event, then P is u.d. iff for each $x \in \Sigma^*$, $P \subseteq D_x(P)$.

<u>Proof</u>: The forward implication is clear from Property 7. Suppose $P \subseteq D_x(P)$ for each $x \in \Sigma^*$. If $z \in P$ then $z \in D_x(P)$ for each $x \in \Sigma^*$. Hence for each $x \in \Sigma^*$ $xz \in P$, or $\Sigma^* z \in P$. This holds for every $z \in P$; hence $\Sigma^* P \subseteq P$. Since $\Lambda \in \Sigma^*$, $P \subseteq \Sigma^* P$, and so $P = \Sigma^* P$. By Property 1 P is u.d.

Q.E.D.

3.2. <u>Definite and Ultimate-Definite GRMs</u>

Although strongly connected automata constitute a proper subclass of the GRMs, for definite and u.d. automata these classes are actually the same. This follows from the presence of an ultimate-definite subset in the accepted event, and the following theorem proves this more general fact.

<u>Theorem 3.2.1</u>: If $R = P + \Sigma^* Q$ where P and Q are arbitrary regular events and $Q \neq \phi$, then A(R) is a GRM iff it is strongly connected.

Proof: Let $A(R) = (Q_A, M, s_0, F)$ be a GRM and suppose $s \in Q_A$. For some $x \in \Sigma^*$, $M(s_0, x) = s$, and for any $y \in Q$ we have $xy \in \Sigma^* Q$; hence $M(s_0, xy) \in F$ and thus $M(s, y) \in F$. So there is a path from s to F, and from each state in F to s_0 since $A(R)$ is a GRM, and from s_0 to every state in Q_A. Hence $R(s) = Q_A$ for each $s \in Q_A$, and $A(R)$ is strongly connected.

The converse is trivial.

Q.E.D.

Thus the above theorem applies to all definite and ultimate-definite events, and shows that no automaton accepting such an event is a Type II GRM.

The next result is a specialization of Theorem 2.3.1 to the u.d. case.

Theorem 3.2.2: If $A(P) = (Q_A, M, s_0, F)$ and $P = \Sigma^* P^t$ is u.d., then $A(P)$ is a GRM iff for each $y \in P$, there is a tape $x \in \Sigma^*$ such that for each suffix $w \neq \Lambda$ of yx, $D_w(P^t) = \phi$.

Proof: If $A(P)$ is a GRM then for each $y \in P$ there is a tape $x \in \Sigma^*$ such that $D_{yx}(P) = P$ by Theorem 2.3.1. But $D_{yx}(P) = P + \bigcup_{w \in S_{yx}} D_w(P^t)$ by Property 7, and $P \cap \left[\bigcup_{w \in S_{yx}} D_w(P^t) \right] = \phi$. Hence $D_w(P^t) = \phi$ for each $w \in S_{yx}$.

Conversely, for $y \in P$ let x be such that $D_w(P^t) = \phi$ for each suffix $w \neq \Lambda$ of yx; that is, $\bigcup_{w \in S_{yx}} D_w(P^t) = \phi$. By Property 7 $D_{yx}(P) = P + \bigcup_{w \in S_{yx}} D_w(P^t)$; hence $D_{yx}(P) = P$ and by Theorem 2.3.1 $A(P)$ is a GRM.

Q.E.D.

Synchronization was defined in Chapter 2, where it was shown that all Type II GRMs are synchronizable. Among the strongly connected machines (i.e., Type I GRMs) some are synchronizable and some are not. For example, machines whose state sets are permuted by every input symbol, that is, $M(Q_A, \sigma) = Q_A$ for each $\sigma \in \Sigma$, are not synchronizable. However, all u.d. machines are synchronizable, as the next theorem shows, and this property is important in characterizing u.d. GRMs. The proof uses the following result of Moore [12].

Lemma 3.2.1: Let $A = (Q_A, M, s_0, F)$ be a reduced f.a. Two distinct states of A can be distinguished by a tape x of length $\leq |Q_A| - 1$.

Theorem 3.2.3: All ultimate-definite automata are synchronizable.

Proof: We prove by contraposition, assuming $A(P)$ is not synchronizable to obtain a contradiction in the construction of P^t. Assume $A(P)$ has n states.

By Theorem 2.4.1 there are two states $s_1, s_2 \in Q_A$ which cannot be merged. Let $x_1, x_2 \in \Sigma^*$ be tapes for which $M(s_0, x_1) = s_1$ and $M(s_0, x_2) = s_2$. There is a tape z_1 of length $\leq n-1$ which distinguishes s_1 and s_2, so that $M(s_1, z_1) = M(s_0, x_1 z_1) \in F$ or $M(s_2, z_1) = M(s_0, x_2 z_1) \in F$ but not both. That is, for $i_1 = 1$ or 2 but not both, $M(s_{i_1}, z_1) = M(s_0, x_{i_1} z_1) \in F$. Hence $x_{i_1} z_1$ has a suffix in P^t. But no suffix of z_1 is in P^t, for then $M(s_1, z_1) \in F$ and $M(s_2, z_1) \in F$ contrary to the selection of z_1. Thus for some non-null suffix $x^1_{i_1}$ of $x_{i_1}, x^1_{i_1} \in P^t$.

$M(s_1, x^1_{i_1})$ and $M(s_2, x^1_{i_1})$ are distinct since s_1 and s_2 cannot be merged. Let z_2 be a tape of length $\leq n-1$ which distinguishes them. Then $M(s_1, x^1_{i_1} z_2) = M(s_0, x_1 x^1_{i_1} z_2) \in F$ or $M(s_2, x^1_{i_1} z_2) = M(s_0, x_2 x^1_{i_1} z_2) \in F$ but not both. Denote the one in F by $M(s_{i_2}, x^1_{i_1} z_2) = M(s_0, x_{i_2} x^1_{i_1} z_2)$. Then $x_{i_2} x^1_{i_1} z_2 \in P$ and, as before, for some non-null suffix $x^2_{i_2}$ of x_{i_2}, $x^2_{i_2} x^1_{i_1} z_2 \in P^t$.

Proceeding in this manner we obtain a list of tapes in P^t as follows:

$$x^1_{i_1} z_1$$

$$x^2_{i_2} x^1_{i_1} z_2$$

$$\vdots$$

$$x^j_{i_j} \cdots x^2_{i_2} x^1_{i_1} z_j$$

$$x^{j+1}_{i_{j+1}} x^j_{i_j} \cdots x^2_{i_2} x^1_{i_1} z_{j+1}$$

$$\vdots$$

Since each z_i has length $\leq n-1$, for some integers m,n such that $m > n$, $z_m = z_n$. But this means $x^n_{i_n} \cdots x^2_{i_2} x^1_{i_1} z_m \in P^t$ and $x^m_{i_m} \cdots x^n_{i_n} \cdots x^2_{i_2} x^1_{i_1} z_m \in P^t$, which is impossible since the former is a proper suffix of the latter.

We conclude that $A(P)$ is synchronizable.

Q.E.D.

Automata which are synchronizable cannot necessarily be synchronized to every state. For example, synchronizable automata with a dead state can be synchronized only to the dead state, as is the case with Type II GRMs. If an automaton A is synchronizable to a state s then it is synchronizable to any state in $R(s)$, for if x synchronizes A to s and $M(s,y) = t$, then xy synchronizes A to t. Of particular interest is the case where A can be synchronized to the start state s_0, or <u>restarted</u>; this is a useful property of many practical machines. Such machines can be synchronized to any state. Ultimate-definite GRMs can be described in terms of this property.

<u>Theorem 3.2.4</u>: Let $A(P) = (Q_A, M, s_0, F)$ be u.d. Then $A(P)$ is strongly connected iff it is synchronizable to s_0.

<u>Proof</u>: All u.d. automata are synchronizable; hence if $A(P)$ is strongly connected it is synchronizable to s_0 since $s_0 \in R(s)$ for each $s \in Q_A$. Conversely, if $A(P)$ is synchronizable to s_0 then $s_0 \in R(s)$ for each $s \in Q_A$, and hence all states are accessible from any state

$s \in Q_A$.

Q.E.D.

We recall from Section 2.1 that a repetitive machine (RM) is a GRM for which there is at least one tape which causes a transition from every final state to the start state. The restartability property of u.d. GRMs leads to the following result.

Theorem 3.2.5: Let $A = (Q_A, M, s_0, F)$ be u.d. Then A is a GRM iff it is an RM.

Proof: By Theorems 3.2.1 and 3.2.4, if A is a GRM then it is restartable; suppose x restarts it. Then $M(Q_A, x) = s_0$, and in particular $M(F,x) = s_0$. Hence A is an RM.

Q.E.D.

The above proof actually indicates a stronger result, namely that any strongly connected synchronizable automaton is an RM.

Cutlip [6] has characterized the tapes which synchronize an ultimate-definite automaton by proving the following: If P is a regular u.d. event, then the tape x synchronizes $A(P)$ iff x is not a prefix of a proper suffix of any tape in P^t. This result, along with the next theorem, characterizes the tapes which synchronize $A(P)$ to the start state.

Theorem 3.2.6: Let x synchronize the u.d. automaton $A(P)$ $= (Q_A, M, s_0, F)$, where $P = \Sigma^* P^t$. Then x restarts $A(P)$ iff no suffix of x of length > 0 is a prefix of any tape in P^t.

Proof: If x restarts $A(P)$ then $D_x(P) = P$ since $M(s_0, x) = s_0$. But $D_x(P) = P + \bigcup_{w \in S_x} D_w(P^t)$ and $P \cap \left[\bigcup_{w \in S_x} D_w(P^t)\right] = \phi$; hence $D_w(P^t) = \phi$ for each $w \in S_x$. That is, no tape in P^t begins with a suffix $w \neq \Lambda$ of x.

Conversely, if no suffix $w \neq \Lambda$ of x is a prefix of any tape in P^t then $D_w(P^t) = \phi$ for each suffix $w \neq \Lambda$ of x, or $\bigcup_{w \in S_x} D_w(P^t) = \phi$. Hence $D_x(P) = P + \bigcup_{w \in S_x} D_w(P^t) = P$ and so $M(s_0, x)$ is indistinguishable from s_0. But since $A(P)$ is reduced, $M(s_0, x) = s_0$. Since x synchronizes $A(P)$, it synchronizes $A(P)$ to s_0.

Q.E.D.

We come now to some results which relate strongly connected u.d. machines to the canonical representation of events.

<u>Theorem 3.2.7</u>: If $A(P) = (Q_A, M, s_0, F)$ is a strongly connected u.d. automaton over the alphabet Σ, then there is a non-empty subset $B \subseteq \Sigma$ such that no tape in P^t begins with a symbol in B.

<u>Proof</u>: If every symbol in Σ is a prefix of a tape in P^t then for each $\sigma \in \Sigma$, $D_\sigma(P^t) \neq \phi$. Then for each $y \in P$ and $x \in \Sigma^*$, there is at least one non-null suffix w of yx such that $D_w(P^t) \neq \phi$. By Theorem 3.2.2 $A(P)$ is not a GRM and hence not strongly connected.

Q.E.D.

For the case of a two-symbol alphabet, if $A(P)$ is strongly connected then every tape in P^t begins with the same symbol. Hence if $\Sigma = \{0,1\}$, the machine for $P = \Sigma^*(1+10)$ is strongly connected, but the machine for $\Sigma^*(00+10)$ is not.

The converse of Theorem 3.2.7 is not true for u.d. events in general, as shown by the machine of Figure 3.2.1 taken from Paz and Peleg [14]. This machine is not strongly connected, although every tape in P^t begins with 1. However, if P^t is finite the converse can be proved.

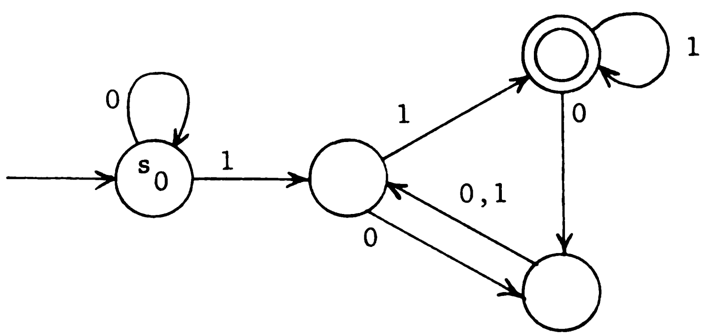


Figure 3.2.1. Machine accepting the event $P = \Sigma^* P^t$, where $\Sigma = \{0,1\}$ and $P^t = 1(00)^*1$.

If $R \subseteq \Sigma^*$ is a finite event, let $\ell(R)$ denote the maximal length of tapes in R. That is, $\ell(R) = \max\{\ell(x) : x \in R\}$.

<u>Theorem 3.2.8</u>: Let $A(P) = (Q_A, M, s_0, F)$ be u.d. over the alphabet Σ.

If P^t is finite, then $A(P)$ is strongly connected iff there is a non-empty subset $B \subseteq \Sigma$ such that no tape in P^t begins with a symbol in B.

Proof: Only the reverse implication must be proved. Suppose no tape in P^t begins with a symbol in B, and let $\ell(P^t) = k$. For each $y \in P$ we will find a tape $x \in \Sigma^*$ such that for each $w \in S_{yx}$, $D_w(P^t) = \phi$, and the proof will be complete by Theorem 3.2.2. Let $x \in B^k$. For each suffix ${}_jx_{k+1}$ of x $(j = 1,\ldots,k)$, $D_{{}_jx_{k+1}}(P^t) = \phi$ since no tape in P^t begins with the first symbol of ${}_jx_{k+1}$. Then for any tape $y \in \Sigma^*$, $D_w(P^t) = \phi$ for each non-null suffix w of yx since no tape in P^t has length exceeding $\ell(x) = k$. Hence for any $y \in P$, any tape $x \in B^k$ has the desired property. By Theorem 3.2.2 $A(P)$ is a GRM and hence strongly connected.

Q.E.D.

The class of u.d. events $P = \Sigma^* P^t$ where P^t is finite is exactly the class of non-initial definite events, which is a proper subclass of both the u.d. events and the definite events; in fact it is the intersection of the two classes.

The next theorem concludes this section.

Theorem 3.2.9: Let $R = P + \Sigma^* Q$ where P is a finite event and R is in canonical form. If $A(R)$ is strongly connected then every tape in P is a proper suffix of a tape in Q.

Proof: Suppose $A(R) = (Q_A, M, s_0, F)$ is strongly connected and let $y \in P$. Then there is a return tape x such that $M(s_0, yx) = s_0$. Hence for any non-negative integer m, $(yx)^m y \in R$. Choose an integer n such that $\ell[(yx)^n y] > \ell(P)$. Then $(yx)^n y \in R$ and $(yx)^n y \notin P$, so $(yx)^n y \in \Sigma^* Q$. Hence $(yx)^n y$ has a suffix in Q. Since $y \in P$, no suffix of y is in Q because R is canonical. Then $(yx)^n y = zw$ where $z \in \Sigma^*$ and $w \in Q$, and since w is not a suffix of y, y is a proper suffix of $w \in Q$.

Q.E.D.

The above theorem applies in particular to composite definite events, and shows that not all such events yield strongly connected machines. For example let $R = 1 + \Sigma^{*}0$.

3.3. Further Structure Properties of Ultimate-Definite Events

Suppose $P = \Sigma^{*}P^{t}$ and $Q = \Sigma^{*}Q^{t}$ are ultimate-definite events. By Property 2 the union and intersection of P and Q are ultimate-definite. If $R = P+Q = \Sigma^{*}R^{t}$ and $T = P \cap Q = \Sigma^{*}T^{t}$, then R^{t} and T^{t} can be found by applying the construction procedure of Property 6. However, the task could be simplified if R^{t} and T^{t} could be determined from P^{t} and Q^{t} only. This problem is the primary concern of the present section.

The first result completely characterizes R^{t} in terms of P^{t} and Q^{t}, where $R = P + Q$.

The following property is an immediate consequence of Definition 3.1.1 and Property 6, but is listed for ease of reference.

Property 12. If $P = \Sigma^{*}W$ is u.d. with canonical representation $P = \Sigma^{*}P^{t}$, then $P^{t} \subseteq W$.

It follows that if $R = \Sigma^{*}R^{t} = \Sigma^{*}P^{t} + \Sigma^{*}Q^{t} = \Sigma^{*}(P^{t} + Q^{t})$, then $R^{t} \subseteq P^{t} + Q^{t}$.

<u>Theorem 3.3.1</u>: Let P and Q by u.d. and $R = P + Q$, with canonical representations $R = \Sigma^{*}R^{t}$, $P = \Sigma^{*}P^{t}$, and $Q = \Sigma^{*}Q^{t}$. If z is a tape, then $z \in R^{t}$ iff either

(1) $z \in P^{t}$ and no proper suffix of z is in Q^{t}, or

(2) $z \in Q^{t}$ and no proper suffix of z is in P^{t}.

<u>Proof</u>: We first prove the implication to the left.

Suppose $z \in P^{t}$ and no proper suffix of z is in Q^{t}. Note that $z \in R$ since $P^{t} \subseteq R$, and hence either $z \in R^{t}$ or a proper suffix of z is in R^{t}. Since $z \in P^{t}$ no proper suffix of z is in P^{t}, and hence none is in $P^{t}+Q^{t}$. But $R^{t} \subseteq P^{t} + Q^{t}$, so z has no proper suffix in R^{t}. Hence $z \in R^{t}$.

If $z \in Q^{t}$ and no proper suffix of z is in P^{t}, a similar argument applies.

The converse is a statement of the form $A \Rightarrow [(B \text{ and } C) \text{ or } (D \text{ and } E)]$, where $A,\ldots,E$ correspond to propositions as follows:

A $\quad z \in R^t$

B $\quad z \in P^t$

C $\quad$ no proper suffix of z is in Q^t

D $\quad z \in Q^t$

E $\quad$ no proper suffix of z is in P^t.

It can be shown that the following logical statement is valid, where "$\sim A$" means "not A."

$\{[(\sim B \text{ and } \sim D) \Rightarrow \sim A] \text{ and } [\sim C \Rightarrow \sim A] \text{ and } [\sim E \Rightarrow \sim A]\} \Rightarrow \{A \Rightarrow [(B \text{ and } C) \text{ or } (D \text{ and } E)]\}$.

To complete the proof of the theorem we will prove the first three implications of the above statement.

$[(\sim B \text{ and } \sim D) \Rightarrow \sim A]$: Suppose $z \notin P^t$ and $z \notin Q^t$. The result $z \notin R^t$ is immediate since $R^t \subseteq P^t + Q^t$.

$[\sim C \Rightarrow \sim A]$: Suppose a proper suffix w of z is in Q^t. Then $w \in Q$ and hence $w \in R$, so either $w \in R^t$ or a proper suffix of w is in R^t. In either case a proper suffix of z would be in R^t, so that $z \notin R^t$.

$[\sim E \Rightarrow \sim A]$: The preceding argument applies with P^t and P substituted for Q^t and Q, respectively.

Q.E.D.

From this result the procedure for constructing R^t from P^t and Q^t, where $R = P + Q$, becomes clear. Beginning with the shortest tapes, place in R^t all tapes in $P^t + Q^t$ which do not have proper suffixes already in R^t. This procedure is in general more efficient than constructing R^t from R. To express the procedure compactly let R_i^t, P_i^t and Q_i^t represent the tapes of length i in R^t, P^t and Q^t, respectively; then

$$R_1^t = P_1^t + Q_1^t$$

$$R_2^t = P_2^t + Q_2^t - (\Sigma Q_1^t + \Sigma P_1^t)$$

$$\vdots$$

$$R_j^t = P_j^t + Q_j^t - \bigcup_{k=1}^{j-1} [\Sigma^{j-k}(P_k^t + Q_k^t)]$$

for $j = 2,3,4,\ldots$, and $R^t = \bigcup_{j=1}^{\infty} R_j^t$.

By Property 12, R^t is contained in $P^t + Q^t$. The conditions for equality follow from Theorem 3.3.1.

Corollary 3.3.1: Let P and Q be u.d. and $R = P + Q$. Then $R^t = P^t + Q^t$ iff no tape in P^t has a proper suffix in Q^t and no tape in Q^t has a proper suffix in P^t.

We now turn to the problem of determining the canonical representation of the intersection of two u.d. events.

Theorem 3.3.2: Let P and Q be u.d. and $T = P \cap Q$, with canonical representations $T = \Sigma^* T^t$, $P = \Sigma^* P^t$, and $Q = \Sigma^* Q^t$. If z is a tape, then $z \in T^t$ iff either

(1) $z \in P^t$ and a suffix of z is in Q^t, or

(2) $z \in Q^t$ and a suffix of z is in P^t.

Proof: We first prove the implication to the left.

Suppose $z \in P^t$ and $z = wx$, where $x \in Q^t$ and $\ell(w) \geq 0$. Then $z \in P$, and $z \in Q$ since a suffix x of z is in Q^t. Hence $z \in P \cap Q = T$. If z has a proper suffix u in T then $u \in P$, and hence either $u \in P^t$ or a proper suffix of u is in P^t. In either case z would have a proper suffix in P^t, which is impossible since $z \in P^t$. Hence z has no proper suffix in T and so $z \in T^t$.

If $z \in Q^t$ and a suffix of z is in P^t, a similar argument applies.

As in Theorem 3.3.1, the converse is a statement of the form $A \Rightarrow [(B \text{ and } C) \text{ or } (D \text{ and } E)]$, where $A,\ldots,E$ correspond to propositions as follows:

A	$z \in T^t$
B	$z \in P^t$
C	a suffix of z is in Q^t
D	$z \in Q^t$
E	a suffix of z is in P^t.

We repeat the procedure used in the proof of Theorem 3.3.1 to complete this proof.

$[(\sim B \text{ and } \sim D) \Rightarrow \sim A]$: Suppose $z \notin P^t$ and $z \notin Q^t$. Either $z \notin T$ or $z \in T$. If $z \notin T$ then clearly $z \notin T^t$, so assume $z \in T = P \cap Q$.

Then z has proper suffixes x in P^t and v in Q^t, and hence $z = wx = uv$ for some non-null tapes w and u. Thus either x is a suffix of v or v is a suffix of x; say $v = yx$ for some (possibly null) tape y. Since $x \in P^t$, $v \in P$. But $v \in Q^t \subseteq Q$, so $v \in P \cap Q = T$. Hence z has a proper suffix v in T, implying $z \notin T^t$.

$[\sim C \Rightarrow \sim A]$: If z has no suffix in Q^t then $z \notin Q$; hence $z \notin T$ and so $z \notin T^t$.

$[\sim E \Rightarrow \sim A]$: If z has no suffix in P^t the preceding argument applies.

Q.E.D.

This theorem yields a procedure for constructing T^t from P^t and Q^t when $T = P \cap Q$. Beginning with the shortest tapes in P^t and Q^t, place in R^t each tape in P^t that has a (not necessarily proper) suffix in Q^t and each tape in Q^t that has a suffix in P^t. Letting T_i^t, P_i^t, and Q_i^t represent the tapes of length i in T^t, P^t, and Q^t, respectively, we can write

$$T_1^t = P_1^t \cap Q_1^t$$

$$T_2^t = P_2^t \cap Q_2^t + P_2^t \cap [\Sigma Q_1^t] + Q_2^t \cap [\Sigma P_1^t]$$

$$T_3^t + P_3^t \cap Q_3^t + P_3^t \cap [\Sigma^2 Q_1^t + \Sigma Q_2^t] + Q_3^t \cap [\Sigma^2 P_1^t + \Sigma P_2^t]$$

$$\vdots$$

$$T_j^t = P_j^t \cap Q_j^t + P_j^t \cap [\bigcup_{k=1}^{j-1} \Sigma^{j-k} Q_k^t] + Q_j^t \cap [\bigcup_{k=1}^{j-1} \Sigma^{j-k} P_k^t]$$

$$\vdots$$

for $j = 2,3,4,\ldots$ Then $T^t = \bigcup_{j=1}^{\infty} T_j^t$.

<u>Corollary 3.3.2</u>: Let P and Q be u.d. and $T = P \cap Q$. Then $T^t \subseteq P^t + Q^t$.

<u>Corollary 3.3.3</u>: Let P and Q be u.d. and $T = P \cap Q$. Then $T^t = P^t + Q^t$ iff $P^t = Q^t$.

<u>Proof</u>: Suppose $T^t = P^t + Q^t$ and let $y \in P^t$. Then $y \in T^t$, and by Theorem 3.3.2 y has a suffix x in Q^t; hence $y = wx$ for some $w \in \Sigma^*$. But $Q^t \subseteq T^t$, so $x \in T^t$. Since y has no proper suffix in

T^t, $w = \Lambda$ and $y = x$. Thus $y \in Q^t$, proving $P^t \subseteq Q^t$. Similarly, $Q^t \subseteq P^t$, and so $P^t = Q^t$.

If $P^t = Q^t$ then $P = Q$ and hence $T = P \cap Q = P = Q$. But $\Sigma^* P^t$ and $\Sigma^* T^t$ are both canonical representations of the same event, so $P^t = T^t$. Similarly, $Q^t = T^t$, and hence $T^t = P^t + Q^t$.

Q.E.D.

For completeness we consider the complement and star of u.d. events. No results were obtained for the concatenation of u.d. events.

If P is u.d. and $R = P^*$, R is u.d. only in the trivial case $R = \Sigma^*$. For, $\Lambda \in P^*$ for any event P and by Property 5, $R = \Sigma^*$ iff $\Lambda \in R$. The canonical representation $\Sigma^* R^t$ of R is given by $R^t = \{\Lambda\}$.

Paz and Peleg [14] proved the following result.

Property 13. If P is u.d. and $\Sigma^* \neq P \neq \phi$, then $\Sigma^* - P$ is not u.d.

From this property it follows that the complement $\Sigma^* - P$ of a u.d. event P is u.d. iff $P = \Sigma^*$ or $P = \phi$.

3.4. Closure Properties of Certain Classes of Events

The class of general repetitive events has been denoted by $\hat{G}$. In addition, let N.I.D., U.D. and DEF denote the classes of non-initial definite, ultimate-definite, and definite events, respectively. The classes N.I.D. $\cap \hat{G}$, U.D. $\cap \hat{G}$, and DEF $\cap \hat{G}$ will be investigated for closure under the standard event operations.

Recall that a non-initial definite event P is an event that can be written $P = \Sigma^* Q$ where Q is finite. Thus N.I.D. is a proper subclass of U.D., and is also a proper subclass of DEF.

Theorem 3.4.1: N.I.D. $\cap \hat{G}$ is not closed under union, dot (concatenation), intersection, complementation or star.

Proof: We provide a counterexample for each case. $\Sigma = \{0,1\}$.

Union: Let $P = \Sigma^* 010$ and $Q = \Sigma^* 11$. Then $P + Q = \Sigma^*(010+11)$. By Theorem 3.2.8 P and Q are in $\hat{G}$ but $P + Q$ is not.

Dot: Let $P = \Sigma^* 1$ and let $R = P^2 = (\Sigma^* 1)(\Sigma^* 1) = \Sigma^*(1\Sigma^* 1)$. Constructing the canonical representation $R = \Sigma^* R^t$ we see that $R^t = 10^* 1$, which is infinite; hence $R \notin$ N.I.D.

Intersection: Let $P = \Sigma^*(01+010)$ and $Q = \Sigma^*(10+101)$. Both events are in $\hat{G}$ by Theorem 3.2.8 and both are in canonical form, with $P^t = 01+010$ and $Q^t = 10+101$. Applying Theorem 3.3.2 we obtain $P \cap Q = \Sigma^*(010+101)$, which by Theorem 3.2.8 is not in $\hat{G}$.

Complementation: Let $P = \Sigma^* 11$. The complement is $\Sigma^* - P = \Sigma^*(10+01+00) + \Lambda + 1 + 0$, which is not in N.I.D.

Star: By Property 4, if P is u.d. then $P^* = \Lambda + P$. This also applies to the non-initial definite events. Let $P = \Sigma^* 1$. Then $P^* = \Lambda + \Sigma^* 1$, which is not in N.I.D.

Q.E.D.

<u>Theorem 3.4.2</u>: U.D. $\cap\ \hat{G}$ is not closed under union, dot, intersection, complementation or star.

<u>Proof</u>: The counterexamples of the preceding theorem apply here for union, intersection, complementation and star.

To see that U.D. $\cap\ \hat{G}$ is not closed under concatenation, consider the events $P = \Sigma^*[00(11)^*1]$ and $Q = \Sigma^* 1$. Both events are u.d., as is the event $R = PQ$. Q is in U.D. $\cap\ \hat{G}$ by Theorem 3.2.8. Construction of the machines for P and R shows that P is a GRE but R is not.

Q.E.D.

<u>Theorem 3.4.3</u>: DEF $\cap\ \hat{G}$ is not closed under union, dot, intersection, or star.

<u>Proof</u>:

Union, dot, intersection: The counterexamples of Theorem 3.4.1 apply. Note that in the case of the dot, R is not in DEF because R^t is infinite.

Star: Let $P = \Sigma^* 11$. By Theorem 3.2.8 $P \in \hat{G}$. By Property 4, $P^* = \Lambda + P$, and construction of the machine for P^* shows that P^* is not a GRE and hence not in $\hat{G}$.

Q.E.D.

The question of closure of DEF $\cap\ \hat{G}$ under complementation has not been resolved. If $P \in$ DEF $\cap\ \hat{G}$, then $\Sigma^* - P \in \hat{G}$ by Theorem 2.5.1.

Hence we must show only that $\Sigma^* - P \in$ DEF. This seems to be a reasonable conjecture.

The above results point up some rather disappointing features of these classes of events. Theorem 3.2.1 shows that the classes N.I.D. $\cap \hat{G}$, U.D. $\cap \hat{G}$ and DEF $\cap \hat{G}$ consist of exactly those events in N.I.D., U.D. and DEF, respectively, that yield strongly connected machines. These results show, for example, that a machine that is equivalent to two strongly connected definite machines operating in parallel (accepting the union of the events accepted by each parallel machine) is not necessarily strongly connected.

In Section 2.5, Theorem 2.5.1 was stated without proof. We now restate the theorem and prove it.

Theorem 2.5.1: Neither $\hat{R}$ nor $\hat{G}$ is closed under union, dot, intersection, complementation or star.

Proof:

Union, intersection: The counterexamples of Theorem 3.4.1 apply for both $\hat{R}$ and $\hat{G}$.

Dot: The counterexample of Theorem 3.4.2 applies.

Complementation: Observe that, for any event P, if $A(P) = (Q_A, M, s_0, F)$, then $A(\Sigma^* - P) = (Q_A, M, s_0, Q_A - F)$. That is, the machine for the complement of P is just the machine for P with final states labelled differently. Let $P = 1(111)^*$. $A(P)$ is a Type II GRM as well as an RM, and since the dead state is in $Q_A - F$, $A(\Sigma^* - P)$ is neither in $\hat{R}$ nor $\hat{G}$.

Star: The counterexample of Theorem 3.4.3 applies.

Q.E.D.

We close this section with a result on the union of ultimate-definite GREs.

Theorem 3.4.4: Let $P, Q \in$ U.D. $\cap \hat{G}$ and let $R = P + Q$. Then $R \in \hat{G}$ iff for each $y \in R$ there is a tape $x \in \Sigma^*$ such that for each non-null suffix w of yx, $D_w(P^t) \subseteq Q$ and $D_w(Q^t) \subseteq P$.

Proof: If $R \in \hat{G}$, then by Theorem 3.2.2 for each $y \in R$ there is a tape $x \in \Sigma^*$ such that $D_{yx}(R) = R$. But $D_{yx}(R) = D_{yx}(P + Q)$

$= P + \bigcup_{w\in S_{yx}} D_w(P^t) + Q + \bigcup_{w\in S_{yx}} D_w(Q^t) = R$. If $z \in \bigcup_{w\in S_{yx}} D_w(P^t)$ then $x \notin P$ (Property 7). But $z \in R$, so $z \in Q$. Hence $\bigcup_{w\in S_{yx}} D_w(P^t) \subseteq Q$, and similarly $\bigcup_{w\in S_{yx}} D_w(Q^t) \subseteq P$.

Suppose that for each $y \in R$ there is a tape $x \in \Sigma^*$ such that for each $w \in S_{yx}$, $D_w(P^t) \subseteq Q$ and $D_w(Q^t) \subseteq P$. That is, $\bigcup_{w\in S_{yx}} D_w(P^t) \subseteq Q$ and $\bigcup_{w\in S_{yx}} D_w(Q^t) \subseteq P$. This implies that $D_{yx}(R) = R$, since $D_{yx}(R)$ $= D_{yx}(P + Q) = P + \bigcup_{w\in S_{yx}} D_w(P^t) + Q + \bigcup_{w\in S_{yx}} D_w(Q^t) = P + Q = R$. By Theorem 3.2.2 R is a GRE.

Q.E.D.

4. ZEIGER'S CASCADE DECOMPOSITION OF AUTOMATA

Several authors have contributed methods of decomposing finite-state machines into an interconnected network of "simpler" machines. Major contributions have been made by Hartmanis and Stearns, Krohn and Rhodes, and Zeiger. The decomposition theory of Hartmanis and Stearns [7] involves partitions on the state set and uses the pair algebra which they developed. Krohn and Rhodes [9, 10] use semigroup theory to show that a finite-state machine can be realized as a cascade connection of two-state machines with no non-trivial permutations and permutation machines of a special type.

In 1965 Zeiger [16] presented a scheme for decomposing an automaton into a cascade of permutation-reset (P-R) machines, and proved the Krohn-Rhodes result by showing that a P-R machine can be decomposed into a cascade of the two basic machines of Krohn and Rhodes. The 1965 paper contained some errors, and was subsequently revised and published in 1967 [17]. This presentation also contained some important discrepancies. A few corrections were published [18], not all of which were correct, and F. Cutlip has suggested corrections for these and the remaining known errors [4].

Some interesting questions arising from the Zeiger decomposition theory are answered in this chapter. In the main these questions concern relations between properties of a machine and the appearance of resets and permutations in the corresponding Zeiger cascade. An application to the decomposition of definite automata is presented.

4.1 The Corrected Decomposition Procedure

To establish notation and to make Chapter 4 self-contained, we present in this section the corrected decomposition procedure of Zeiger. Some of the notation of the previous chapters is changed in order to be consistent with Zeiger's notation.

M, N, K, and L denote finite automata, or machines. The state set of M is Q_M, and the input set is I_M. I_M^* is the set of all finite sequences obtained by concatenating symbols from I_M, including the null sequence Λ, and is the set of all input sequences available to M. If $\sigma \in I_M$, then σ_M is the transformation of Q_M defined as follows: if $B \subseteq Q_M$, then $\sigma_M(B) = M(B,\sigma) = \{s \in Q_M\colon M(t,\sigma) = s$ for some $t \in B\}$, where $M(\cdot,\cdot)$ is the state transition function of M. The range of σ_M is written ran σ_M. If $\sigma,\tau \in I_M$, then the composition of σ_M and τ_M is $\sigma_M\tau_M(B) = \sigma_M[M(B,\tau),\sigma] = M(B,\tau\sigma) = (\tau\sigma)_M(B)$, and S_M denotes the semigroup of transformations generated by elements of I_M under this composition operation.

If A is a set, E_A denotes the identity map on A. A <u>reset</u> of machine M is a map w that takes all states in Q_M to a single state; that is, for some $s \in Q_M$ and for all $t \in Q_M$, $w(t) = s$. The extended semigroup $\tilde{S}_M$ is defined by $\tilde{S}_M = S_M \cup E_{Q_M} \cup \{w\colon w$ is a reset of $M\}$. If $w \in \tilde{S}_M$ is identity map on $B \subseteq Q_M$, we write $w(B) = B(\text{id})$.

A <u>cover</u> C for machine M is a nonempty collection of nonempty subsets of Q_M such that for each $w \in \tilde{S}_M$ and $R \in C$, $w(R)$ is a subset of an element of C. Observe that every state of Q_M is in some cover element. Cover elements and state subsets will usually be denoted by capital letters.

If C is a cover for M and N is a machine, then <u>N tells where M is in C</u> means that (1) $I_N = I_M$ and (2) there is a map Z_N of Q_N onto C such that for each $\sigma \in I_M$ and $s \in Q_N$, $\sigma_M(Z_N(s)) \subseteq Z_N(\sigma_N(s))$.

Let K,L, and N be machines. Then $N \in K \rightarrow L$ is read "N is a series composition of K followed by L" and means (1) $Q_N \subseteq Q_K \times Q_L$, $I_N \subseteq I_K$, and $I_L \subseteq I_K \times Q_K$, and (2) for each $\sigma \in I_N$ and $(p,r) \in Q_N$, $\sigma_N(p,r) = (\sigma_K(p),(\sigma,p)_L(r))$. See Figure 4.1.1.

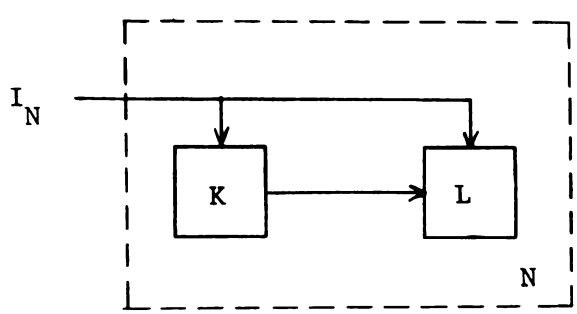


Figure 4.1.1. Illustration of $N \in K \rightarrow L$.

If A is a collection of sets, then $\max A$ denotes the set of all elements of A which are not contained in other elements of A.

A permutation-reset (P-R) machine L is a machine for which each input $\sigma \in I_L$ either resets L ($\sigma_L(Q_L)$ is a single state) or permutes the states of L ($\sigma_L(Q_L) = Q_L$).

The strategy for decomposing machine M is as follows. First obtain a nontrivial cover C for M and construct a P-R machine K which tells where M is in C. Then refine the cover C by selecting certain elements of C and replacing them with certain smaller subsets of Q_M to obtain a refined cover C'. Construct a P-R machine L so that $N \in K \to L$ and N tells where M is in C. Continue this process until a cover consisting only of singletons is reached.

Theorem 4.1.1 (Zeiger's Proposition 1) shows how the first cover and P-R machine of a cascade decomposition of M are constructed; Theorem 4.1.2 (Zeiger's Proposition 2) gives the induction step, showing that successive refined covers and corresponding P-R machines can be constructed to complete the cascade.

Theorem 4.1.1 (Zeiger): For each machine M there is a cover C, not containing Q_M, and a machine N for which (1) N tells where M is in C, (2) for each $\sigma \in I_N$, ran σ_N is either Q_N or a singleton, and (3) the permutations of Q_N are uniquely determined.

Proof: Let $C = \max(\{w(Q_M): w \in \tilde{S}_M\} - Q_M)$. Let $Q_N = C$ and $Z_N = E_C$. The transitions of N are defined as follows. Let $\sigma \in I_N = I_M$.

(1) If $\sigma_M(Q_M) \neq Q_M$, then $\sigma_M(Q_M) \subseteq R' \in C$ for some R'. For each $R \in C$ let $\sigma_N(R) = R'$. If more than one R' has this property, choose any one. Note that σ_N resets N to state $R' \in Q_N$.

(2) Suppose $\sigma_M(Q_M) = Q_M$; that is, σ_M permutes Q_M. If $R \in C$ then $\sigma_M(R) \subseteq T \in C$. Since σ_M is a permutation it has an inverse; thus $\sigma_M^{-1}(T)$ contains R. But this cannot be a proper containment since $R \in C$ and C is max of a collection of subsets. Hence $\sigma_M^{-1}(T) = R$, and $\sigma_M(R) = T \in C$, showing that σ_M maps elements of C onto elements of C.

Since σ_M is invertible on Q_M, it is invertible on subsets of Q_M, and in particular on C. Hence σ_M is 1-1 on C, and so permutes C.

For $R \in C$ let $\sigma_N(R) = R'$ if and only if $\sigma_M(R) = R'$. Since σ_M permutes C, σ_N permutes C, and so permutes the states of N.

It is clear that N is a P-R machine and the permutations of Q_N are uniquely determined. Since Z_N is identity on C, the condition "N tells where M is in C" is satisfied.

Q.E.D.

Corollary 4.1.1 (Zeiger): The group of permutations in S_N is a homomorphic image of a subgroup of S_M.

Proof: Define $G_M \subseteq S_M$ by $G_M = \{x_M \in S_M: x_M(Q_M) = Q_M\}$ and define $G_N \subseteq S_N$ by $G_N = \{x_N \in S_N: x_N(Q_N) = Q_N\}$ or, equivalently, $G_N = \{x_N \in S_N: x_N(C) = C\}$. It is easily verified that G_M and G_N are groups. The map $f: G_M \rightarrow S_M$ defined by $f(x_M) = x_N$ for $x_M \in G_M$ is the desired homomorphism. Note that f is in general many-to-one since two input sequences to M may permute Q_M differently but have the same effect on C.

Q.E.D.

Figure 4.1.2 shows the result of applying Theorem 4.1.1 to a machine M. Note that the second box, representing "what is left" of M, is unspecified.

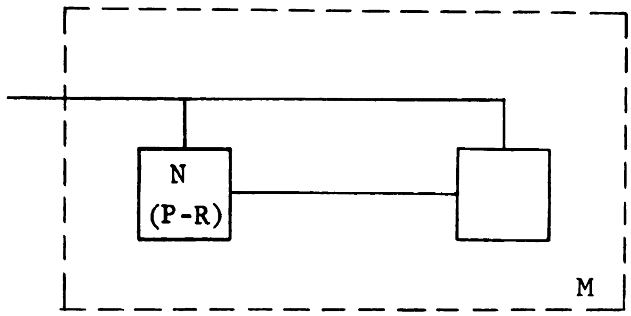


Figure 4.1.2. The first step in decomposing M.

THES

The following lemmas are used in the proof of Theorem 4.1.2.

Definition 4.1.1: Let C be a cover for M. Elements P and R of C are said to be similar if there are transformations x_M and y_M in $\tilde{S}_M$ such that $x_M(P) = R$ and $y_M(R) = P$. An element R of C is initial in C if it is not the image under transformations from $\tilde{S}_M$ of any element of C except those similar to it.

Similarity is an equivalence relation on C. If $R \in C$ is initial, then so is every element in the similarity class of R. Such classes are called initial similarity classes.

Lemma 4.1.1: If C is a cover for M then C has an initial similarity class among the elements of C of maximal cardinality.

Proof: Let D_1 be a similarity class whose elements are of maximal cardinality. If D_1 is not initial, then another similarity class $D_2 \neq D_1$ has elements with images in D_1 under appropriate transformations from $\tilde{S}_M$. If D_2 is not initial, select D_3 in a similar fashion, and so on. Since the elements of D_i are not similar to any elements in D_j for $i \neq j$, no D_i can appear twice in the sequence. Since there are a finite number of similarity classes in C, the sequence must terminate in an initial similarity class.

Q.E.D.

Lemma 4.1.2: Let C be a cover for M having initial similarity class D. If $P \in D$ and $R \in D$ then there are transformations v_P^R and v_R^P in $\tilde{S}_M$ with the properties (1) $v_P^R(P) = R$ and $v_R^P(R) = P$, and (2) $v_P^R \cdot v_R^P(R) = R(id)$ and $v_R^P \cdot v_P^R(P) = P(id)$.

Proof: By similarity, for some $x_M, y_M \in \tilde{S}_M$, $x_M(P) = R$ and $y_M(R) = P$. Thus $y_M x_M(P) = P$ and $x_M y_M(R) = R$. There are integers m,n for which $(y_M x_M)^m(P) = P(id)$ and $(x_M y_M)^n(R) = R(id)$, and hence $(y_M x_M)^{mn}(P) = P(id)$ and $(x_M y_M)^{mn}(R) = R(id)$. Let $v_P^R = x_M$ and $v_R^P = (y_M x_M)^{mn-1} y_M$; these clearly have the desired properties.

Q.E.D.

We now can present Zeiger's Proposition 2. See Figure 4.1.3.

THE

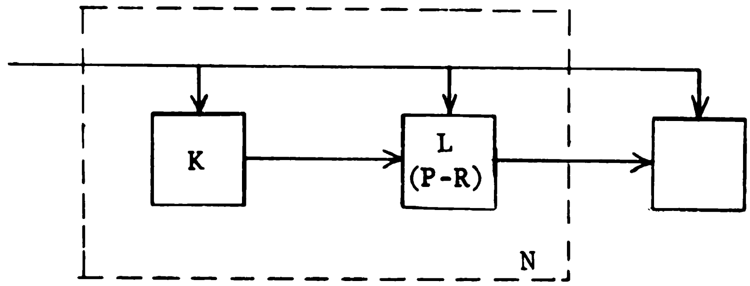


Figure 4.1.3. The induction step in decomposing M.

Theorem 4.1.2 (Zeiger): If K and M are machines, C is a cover for M not consisting entirely of singleton sets, and K tells where M is in C, then there are machines N and L and a cover C' for M for which:

(1) C' is a proper refinement of C, i.e., if $R \in C'$ then R is contained in an element of C, and for some $T \in C$, T is not a subset of any element of C'.

(2) N tells where M is in C'.

(3) $N \in K \to L$.

(4) For each $(\sigma,p) \in I_L$, ran $(\sigma,p)_L$ is either Q_L or a singleton.

(5) There is a subset $B \subseteq Q_M$ for which there is a homomorphism from the group of all permutations of B produced by state transformations of M onto the group in S_L. (It follows that the group in S_L is a homomorphic image of a subgroup of S_M.)

Proof: To construct C', choose an initial similarity class D among the elements of C of maximal cardinality. C' is obtained from C by replacing each $P \in D$ with a collection of subsets of P to be specified. Denoting that collection $R(P)$ (the replacement set for P), define $R(P) = \max \{w(T): w \in \tilde{S}_M, T \in C, \text{ and } w(T) \text{ is a proper subset of } P\}$. Then $C' = (C-D) \cup [\bigcup_{P\in D} R(P)]$. Clearly C' is a proper refinement of C (if C consists only of singletons the cascade in complete and C' is empty).

THE

Machines N and L are constructed as follows. Choose a fixed element U of D, which will be called the reference element of D. For each $P \in D$ select transformations v_U^P and v_P^U according to Lemma 4.1.2. (Note that if $A \in R(U)$ and $P \in D$, then $v_U^P(A) \in R(P)$; for if A is properly contained in $A' \in R(P)$ then $v_P^U(A') \supsetneq A$ since v_P^U is 1-1, implying that A was not max among the subsets from which R(U) was chosen.) Let $Q_L = \{R \in C' : R \subset U\}$; that is, to each element of R(U) we correspond a state of Q_L. Let $I_L = I_K \times Q_K$, $Q_N = Q_K \times Q_L$ and $I_N = I_K = I_M$. Let Z_N map Q_N onto C' so that for each $(p,r) \in Q_N$,

(1) If $Z_K(p) = P \notin D$, then $Z_N(p,r) = Z_K(p)$.

(2) If $Z_K(p) = P \in D$, then $Z_N(p,r) = v_U^P(r)$, where the symbol r denotes both a state of L and an element of C'.

One further lemma is needed before the transitions of N can be specified.

Lemma (Cutlip): Let $P \in C$ and $S \in D$. If $\sigma_M(P)$ is properly contained in S, there is an element A in R(U) such that $v_S^U \sigma_M(P) \subseteq A$.

Proof: If $\sigma_M(P) \subsetneq S$ then $v_S^U \sigma_M(P) \subsetneq U$ since v_S^U maps S one-to-one onto U. Hence $v_S^U \sigma_M(P)$ is among the subsets of U whose max is taken to get R(U).

Q.E.D.

To specify the transitions of N, for each $\sigma \in I_N$ and $(p,r) \in Q_N$, define $\sigma_N(p,r) = (s,t)$ by:

(1) $s = \sigma_K(p)$.

(2) If $Z_K(s) = S \in C \cap C'$, let t = "don't care".

(3) If $Z_K(p) = P \in C \cap C'$ and $Z_K(s) = S \in D$, choose t so that $v_S^U \sigma_M(P) \subseteq t$.

(4) If $Z_K(p) = P \in D$ and $Z_K(s) = S \in D$, choose t so that $v_S^U \sigma_M v_U^P(r) \subseteq t$.

Again, the symbols r and t refer both to states of L and corresponding elements of C' in R(U). In step 2, the next state of L can be arbitrarily chosen. Zeiger chose to let t = r, which means σ_L performs the identity permutation on states of L. The "don't care" specification given here is preferred so that when the terms "permutation" and "reset" are used, only the nontrivial cases in steps 3 and 4 are understood. In step 3, σ_L resets L to state t, and such a state transition specification is allowable in view of the preceding lemma. In step 4, when $v_S^U \sigma_M v_U^P(r) = t$ a permutation of Q_L is produced by σ. In some situations σ_M is not a one-to-one map from P to S, and $v_S^U \sigma_M v_U^P(r)$ is properly contained in t; in this case σ resets L to state t, and this is also an allowable transition in view of the lemma.

These transition specifications for N will be referred to frequently in this chapter as Rules 1, 2, 3, and 4.

This completes the construction of N and L. Conclusions 1, 3 and 4 of the theorem are verified directly. The proofs of conclusions 2 and 5 do not significantly aid understanding of the construction; hence the former is included in the appendix and the latter is omitted.

Q.E.D.

The machine L is called a <u>component machine</u>.

Examples of this decomposition are given by Cutlip in [4].

The next two sections contain the new results of this study.

4.2. <u>Permutations, Resets, and Zeiger Covers</u>

Let M be a machine, and let $C_1, \ldots, C_n$ be a sequence of covers obtained in a cascade decomposition of M as described in the previous section, where C_n consists of singletons. This sequence is not unique in general, since each cover may have more than one initial similarity class available for refinement. Each C_i is called a <u>Zeiger cover</u>, and $C_1, \ldots, C_n$ is called a <u>sequence of Zeiger covers for M.</u> The term <u>Zeiger cover for M</u> refers to an arbitrary cover from any sequence of Zeiger covers for M.

In this section the properties of Zeiger covers for M and subsets of Q_M which have special significance in a cascade decomposition of M are investigated. Necessary and sufficient conditions for an input to cause resets or identity permutations of the component machines are obtained.

Throughout this section the following situation is assumed.

THE

$C_1,\ldots,C_n$ is a sequence of Zeiger covers for M, with initial similarity classes $D_1,\ldots,D_{n-1}$ and corresponding component machines $L_1,\ldots,L_n$. The cascade connection of $L_1,\ldots,L_n$ is a Zeiger cascade for M. K_i or N_i is the cascade of $L_1,\ldots,L_i$. For economy of space these conditions will not be written into every lemma and theorem.

The first result shows the structure of initial similarity classes.

Theorem 4.2.1: Let $C = \{S_1,\ldots,S_m\}$ be a Zeiger cover for M with initial similarity class $D = \{S_1,\ldots,S_k\}$, $k \le m$, and reference element S_1. For each $i \le k$ let the replacement set for S_i be $R(S_1) = \{S_{i1},S_{i2},\ldots,S_{ir_i}\}$.

Then $r_1 =\ldots= r_k = r$, and the labels S_{ij} can be assigned so that $v_{S_1}^{S_i}(S_{1h}) = S_{ih}$ and $v_{S_i}^{S_1}(S_{ih}) = S_{1h}$ for each $i \le k$ and $h \le r$.

Proof: Let $R(S_1) = \{S_{11},S_{12},\ldots,S_{1r_1}\}$ be any labelling of $R(S_1)$. For each $i \le k$ select $v_{S_1}^{S_i}$ and $v_{S_i}^{S_1}$ according to Lemma 4.1.2. To assign labels to $R(S_i)$ define $S_{ij} = v_{S_1}^{S_i}(S_{1j})$ for $j = 1,\ldots,r_1$ and note that $v_{S_i}^{S_1}(S_{ij}) = v_{S_i}^{S_1}v_{S_1}^{S_i}(s_{1j}) = S_{1j}$. We must show that $S_{ij} \in R(S_i)$ for each j and that $v_{S_1}^{S_i}$ maps $R(S_1)$ onto $R(S_i)$.

S_{ij} is contained in some element T of $R(S_i)$, and $T = w(S)$ for some $w \in \tilde{S}_M$ and $S \in C$ by definition of $R(S_i)$. Now $S_{1j} = v_{S_i}^{S_1}(S_{ij}) \subseteq v_{S_i}^{S_1}(T) = v_{S_i}^{S_1}w(S)$. But $v_{S_i}^{S_1}w \in \tilde{S}_M$, so $v_{S_i}^{S_1}w(S)$ is among the subsets of Q_M whose max is taken to get $R(S_1)$. But $S_{1j} \in R(S_1)$, so $S_{1j} = v_{S_i}^{S_1}(T)$ is required. Thus

$$S_{ij} = v_{S_1}^{S_i}(S_{1j}) = v_{S_1}^{S_i}v_{S_i}^{S_1}(T) = T \in R(S_i).$$

To show that $v_{S_1}^{S_i}: R(S_1) \to R(S_i)$ is onto, let T be any element of $R(S_i)$. Then for some $h \le r_1$ $v_{S_i}^{S_1}(T) \subseteq S_{1h} \in R(S_1)$.

THE

Thus $T = v_{S_1}^{S_i} v_{S_i}^{S_1}(T) \subseteq v_{S_1}^{S_i}(S_{1h}) = S_{ih}$, and so $T = S_{ih}$ since $T \in R(S_i)$.

Q.E.D.

It is not necessarily true that $v_{S_i}^{S_j}(S_{ih}) = S_{jh}$ for every $i,j \leq k$ and $h \leq r$. However, $v_{S_i}^{S_j}$ can be selected to have this property. Let $v_{S_i}^{S_j} = v_{S_1}^{S_j} v_{S_i}^{S_1}$. Then $v_{S_i}^{S_j}(S_{ih}) = v_{S_1}^{S_j} v_{S_i}^{S_1}(S_{ih}) = v_{S_1}^{S_j}(S_{1h}) = S_{jh}$, and it is easily verified that $v_{S_j}^{S_i} v_{S_i}^{S_j}(S_i) = S_i(\text{id})$.

Zeiger shows that the permutations of the states of a component machine are caused by permutations of subsets of states of the original machine. The following results characterize these permuted state subsets and show that some appear as elements in Zeiger covers.

<u>Notation</u>: Let $x_M \in \tilde{S}_M$ and $B \subseteq Q_M$. If $x = \sigma_1\sigma_2\ldots\sigma_k \in I_M^*$ then $x_M(B) = (\sigma_k)_M \ldots (\sigma_1)_M(B) = M(B,x)$, where $M(\cdot,\cdot)$ is the state transition function and $M(B,x) = \{s \in Q_M\colon \text{for some } t \in B, M(t,x) = s\}$. Note that $(\sigma_1)_M$ is applied first to B. If i is a positive integer, then $x_M^i(B) = M(B,x^i)$, where x^i is x concatenated i times. Define $x_M^o(B) = \Lambda_M(B) = B(\text{id})$, where Λ is the null tape.

If x_M is a reset such that $x \notin I_M^*$, let $x_M^i(B) = x_M(Q_M)$ for $i = 1,2,\ldots.$

<u>Definition 4.2.1</u>: Let $x \in I_M^*$. A subset B of Q_M is called a <u>minimal x-permuted subset of Q_M</u> if and only if $x_M(B) = B$ and $x_M(A) \neq A$ for all proper subsets A of B.

If x_M is a permutation on Q_M $(x_M(Q_M) = Q_M)$ then the minimal x-permuted subsets of Q_M are the orbits of the permutation.

<u>Lemma 4.2.1</u>: Suppose $B \subseteq Q_M$ and $x \in I_M^*$. Then $x_M(B) = B$ if and only if B is the union of minimal x-permuted subsets of Q_M.

<u>Proof</u>: If $x_M(B) = B$ and $A_B = A \cap B \neq \emptyset$, where A is a minimal x-permuted subset of Q_M, then $x_M(A_B) \subseteq B$. But $x_M(A) = A$, so $x_M(A_B) \subseteq A$, yielding $x_M(A_B) \subseteq A \cap B = A_B$. Since A is finite and x_M is one-to-one on A, $x_M(A_B) = A_B$. Thus $A_B = A$ since A has no

THE

proper subsets permuted by x_M, and $A \subseteq B$ since $A = A \cap B$. That is, any minimal x-permuted subset of Q_M is either contained in B or disjoint from B. Noting that every element of B is an element of some minimal x-permuted subset, we conclude that B is the union of such subsets.

Conversely, if $B = B_1 \cup \ldots \cup B_k$ where each B_i is a minimal x-permuted subset, then $x_M(B) = x_M(B_1 \cup \ldots \cup B_k) = x_M(B_1) \cup \ldots \cup x_M(B_k) = B_1 \cup \ldots \cup B_k = B$.

Q.E.D.

Definition 4.2.2: For each $x_M \in \tilde{S}_M$ define $S_{x^i} = x_M^i(Q_M)$ for $i = 1,2,\ldots$. If $S_{x^i} = S_{x^{i+1}}$ for some i, let S_{x^i} be called S_{x^∞}.

Lemma 4.2.2: For each $x_M \in \tilde{S}_M$ there is an integer k such that $S_{x^\infty} = S_{x^k} = S_{x^{k+j}}$ for all integers $j \geq 0$, and $S_{x^i} \subsetneq S_{x^{i-1}}$ for $i \leq k$.

Proof: Since $x_M(Q_M) \subseteq Q_M$, $x_M^i(Q_M) = x_M^{i-1}[x_M(Q_M)] \subseteq x_M^{i-1}(Q_M)$. Repeated application of x_M to Q_M yields a nested sequence of subsets $Q_M, S_x, S_{x^2}, S_{x^3}, \ldots$ of Q_M. Let S_{x^k} be the subset of lowest index with the property $S_{x^k} = S_{x^{k+1}}$. Such a subset exists since Q_M is finite. Then $x_M(S_{x^k}) = x_M(S_{x^{k+1}})$, or $S_{x^{k+1}} = S_{x^{k+2}}$, and clearly $S_{x^k} = S_{x^{k+j}}$ for all $j \geq 0$. For $i \leq k$, S_{x^i} is properly contained in $S_{x^{i-1}}$ because of the choice of k.

Q.E.D.

Lemma 4.2.3: For each $x_M \in \tilde{S}_M$, S_{x^∞} is the union of all the minimal x-permuted subsets of Q_M.

Proof: Since $x_M(S_{x^\infty}) = S_{x^\infty}$, S_{x^∞} is the union of minimal x-permuted subsets by Lemma 4.2.1. If $B \subseteq Q_M$ is any minimal x-permuted subset and k is an integer such that $S_{x^k} = S_{x^\infty}$, then $B = x_M^k(B) \subseteq x_M^k(Q_M) = S_{x^\infty}$. Hence every minimal x-permuted subset of Q_M is contained in S_{x^∞}.

Q.E.D.

THE

We can now relate the subsets S_{x^∞} to the Zeiger covers for M. The effect of S_{x^∞} on component machines will be shown later.

Theorem 4.2.2: Let $x_M \in \tilde{S}_M$. If $x_M(Q_M) \neq Q_M$, there is a cover having S_{x^∞} as an element.

Proof: If S_{x^∞} is a singleton, then $S_{x^\infty} \in C_n$. If not, S_{x^∞} is contained in at least one cover element, namely C_1. Let $C_u \in \{C_1,\ldots,C_n\}$ be the cover of largest index having an element B containing S_{x^∞}. Then $B \in D_u$; otherwise $B \in C_{u+1}$.

For each positive integer i let $x_M^i(B) = B_i$. Then $S_{x^\infty} \subseteq B_i$ since $S_{x^\infty} \subseteq B$. Hence B_i is contained in an element T_i of D_u; otherwise $T_i \in C_{u+1}$, contrary to the choice of C_u.

If B_i is properly contained in T_i then B_i is among the subsets from which $R(T_i)$ is chosen, and consequently S_{x^∞} is a subset of an element of $R(T_i)$ since $S_{x^\infty} \subseteq B_i$. This implies S_{x^∞} is contained in an element of C_{u+1}, contrary to the choice of C_u. Hence $B_i = T_i \in D_u$ for each i.

Since D_u has a finite number of elements, there are integers k and m with $m \geq k$ such that $x_M(B_m) = B_k$. Then $x_M^{m-k}(B_k) = B_m$, so $x_M^{m-k+1}(B_k) = x_M(B_m) = B_k$. Let r be any integer such that $x_M^r(Q_M) = S_{x^r} = S_{x^\infty}$, and choose j such that $(m-k+1)j \geq r$. Then $x_M^{(m-k+1)j}(Q_M) = S_{x^\infty}$, and hence $B_k = x_M^{(m-k+1)j}(B_k) \subseteq S_{x^\infty}$. But $S_{x^\infty} \subseteq B_i$ for each i, so $B_k = S_{x^\infty}$ and hence $S_{x^\infty} \in C_u$.

Q.E.D.

It is possible to construct a sequence $C_1,\ldots,C_n$ of Zeiger covers for M without constructing the corresponding component machines $L_1,\ldots,L_n$. What information about the permutations and resets in the component machines can be obtained from the covers? This question is examined in the next few results.

Let K_i be the cascade of machines $L_1,\ldots,L_i$. Recall that an input to L_{i+1} is a pair (p,σ) where $p \in Q_{K_i}$ and $\sigma \in I_M$. The symbol σ is said to cause a permutation of L_{i+1}^i if for some p, THE1

$(p,\sigma)_{L_{i+1}}(Q_{L_{i+1}}) = Q_{L_{i+1}}$. Observe that σ might cause a permutation of L_{i+1} when K_i is in state p, but reset L_{i+1} when K_i is in a state other than p. The phrase $\underline{L_{i+1}}$ <u>has permutations</u> means that some symbol $\sigma \in I_M$ causes a permutation of L_{i+1}.

In Theorem 4.1.2 the replacement set R(P) for an element P of initial similarity class D was defined. We extend this to define the replacement set for D as $R(D) = \bigcup_{P \in D} R(P)$.

The next lemma follows directly from the procedure for constructing a Zeiger cascade for M given in Theorem 4.1.2, but is presented here because of its importance in the next several results.

<u>Lemma 4.2.4</u>: If $B,B' \in D_i$ and $\sigma_M(B) = B'$, then σ cause a permutation of L_{i+1}.

<u>Proof</u>: Let K_i be the cascade of $L_1,\ldots,L_i$. Since K_i tells where M is in C_i, there are states p and s of Q_{K_i} such that $Z_{K_i}(p) = B$ and $Z_{K_i}(s) = B'$. Since $B,B' \in D_i$, the state transitions of L_{i+1} are determined by Rule 4 of Theorem 4.1.2. But σ_M is a one-to-one map from B to B', so the input (p,σ) to L_{i+1} permutes the states of L_{i+1}. That is, σ causes a permutation of L_{i+1}.

Q.E.D.

<u>Lemma 4.2.5</u>: If $B \in D_i$ and $x_M(B) \in D_i$ for $x_M \in \tilde{S}_M$, then every prefix of x maps B onto an element of D_i.

<u>Proof</u>: B and $x_M(B)$ are both in D_i and hence similar. Let $y_M \in \tilde{S}_M$ be the transformation taking $x_M(B)$ to B; that is, $y_M x_M(B) = (xy)_M(B) = B$. Let w_1 be a prefix of x, so that $x = w_1w_2$, and let $(w_1)_M(B) = B'$. B' is an element of C_i since $(w_1)_M$ is one-to-one and B is of maximal cardinality in C_i. Then $(w_2)_M(B') = (w_2)_M(w_1)_M(B) = (w_1w_2)_M(B) = x_M(B)$ and $y_M(w_2)_M(B') = y_M x_M(B) = B$. Thus B and B' are similar, implying $B' \in D_i$.

Q.E.D.

<u>Theorem 4.2.3</u>: Suppose $x_M \in \tilde{S}_M$. If $S_{x^\infty} \in D_i$ then every symbol embedded in x causes a permutation of L_{i+1}.

<u>Proof</u>: Suppose $x = w_1\sigma w_2$ where $w_1,w_2 \in I_M^*$ and $\sigma \in I_M$. Let

$(w_1)_M(S_{x^\infty}) = B$ and $(w_1\sigma)_M(S_{x^\infty}) = B'$. Then $\sigma_M(B) = B'$, and $B \in D_i$ and $B' \in D_i$ by Lemma 4.2.5. Hence σ causes a permutation of L_{i+1} by Lemma 4.2.4.

Q.E.D.

Theorem 4.2.3 shows that the presence of the subsets S_{x^∞} in initial similarity classes produces permutations in the corresponding component machines. The next theorem shows that non-singleton initial similarity classes produce permutations in their component machines, and that subsequent refinements of these classes give the same result. First a lemma is required.

Lemma 4.2.6: Let C_u and C_r be covers from a sequence $C_1,\ldots,C_n$ of Zeiger covers for M, and let $v > u$. Let $C_u = \{S_1,\ldots,S_m\}$ and $D_u = \{S_1,\ldots,S_k\}$, $k \le m$. For each $i \le k$, let $R(S_i) = \{S_{i1},\ldots,S_{ir}\}$.

If $S_{ij} \in D_v$ for any $i \le k$ and $j \le r$, then $\{S_{1j},\ldots,S_{kj}\} \subseteq D_v$.

Proof: It is sufficient to show that the elements of $\{S_{1j},\ldots,S_{kj}\}$ are similar, and hence when one element appears in an initial similarity class, all do. Choose the transformations $v_{S_1}^{S_i}$ and $v_{S_i}^{S_1}$ for $i \le k$ and assume the elements of $R(D_u)$ are labelled so that $v_{S_1}^{S_i}(S_{1j}) = S_{ij}$. Then $v_{S_i}^{S_1}(S_{ij}) = S_{1j}$, and hence S_{1j} and S_{ij} are similar. By transitivity of similarity, all elements in $\{S_{1j},\ldots,S_{kj}\}$ are similar.

Q.E.D.

Theorem 4.2.4: If D_u is not a singleton, then L_{u+1} has permutations, and if D_v contains an element of $R(D_u)$ for $v > u$ then L_{v+1} has permutations.

Proof: Let B,B' be distinct elements of D_u. Since they are similar, $x_M(B) = B'$ for some $x_M \in \tilde{S}_M$. By Lemma 4.2.5 each prefix of x maps B onto an element of D_u, and hence each symbol embedded in x maps an element of D_u onto another, causing a permutation of L_{u+1} by Lemma 4.2.4.

Lemma 4.2.6 shows that if D_v contains an element of $R(D_u)$

THE

then it contains an element of $R(B)$ for each $B \in D_u$. So if D_u is not a singleton, neither is D_v, and L_{v+1} has permutations by the argument above.

Q.E.D.

A natural question concerns the conditions for obtaining a cascade having no permutations. The next theorem answers this question, and in the following section the events realized by such machines are examined.

Theorem 4.2.5: A Zeiger cascade M has no permutations if and only if $S_{x\infty}$ is a singleton for all x in $\{I_M^* - \Lambda\}$.

Proof: Suppose $S_{x\infty}$ is not a singleton for some $x \in \{I_M^* - \Lambda\}$. If $S_{x\infty} \doteq Q_M$ then the first machine of the cascade has permutations, and if $S_{x\infty} \neq Q_M$ then by Theorem 4.2.3 some component machine has permutations, proving the forward implication by contraposition.

Suppose $S_{x\infty}$ is a singleton for all $x \in \{I_M^* - \Lambda\}$. Let $C_1,\ldots,C_n$ be a sequence of Zeiger covers for M with component machines $L_1,\ldots,L_n$, and let N_i be the cascade of $L_1,\ldots,L_i$ for each $i \leq n$. We prove by induction on the machines $N_1,\ldots,N_n$ that the cascade for M has no permutations.

For each $\sigma \in I_M$, $S_{\sigma\infty}$ is a singleton, so no input symbol permutes Q_M. Hence $N_1 = L_1$ is reset by every input.

Assume N_i is a cascade of resets and "don't cares" and consider transitions $\sigma_{N_{i+1}}(p,r) = (s,t)$ where $Z_{N_i}(p) = P \in D_i$ and $Z_{N_i}(s) = S$. If $S \notin D_i$ then σ produces a "don't care" in L_{i+1}. Suppose $S \in D_i$. Then $\sigma_M(P) \subseteq S$. If the containment is proper, σ produces a reset in L_{i+1}. Suppose $\sigma_M(P) = S$. Since P and S are similar there is a transformation y_M such that $y_M(S) = P$. Then $y_M\sigma_M(P) = (\sigma y)_M(P)$ $= y_M(S) = P$; thus $(\sigma y)_M$ permutes P and P is the union of minimal (σy)-permuted subsets of Q_M. But $S_{(\sigma y)\infty}$ is a singleton and $P \subseteq S_{(\sigma y)\infty}$, so P is a singleton. Hence P could not appear in the initial similarity class D_i. This shows that L_{i+1}, and hence the cascade N_{i+1}, has no permutations, which proves the theorem.

Q.E.D.

We finally turn to the special but important case in which the symbol σ permutes the whole state set. In this case the minimal σ-permuted subsets of Q_M are the orbits of the permutation. It is clear that σ will cause a permutation of L_{i+1} whenever the state of N_i corresponds to a D_i element, say B, for σ maps B one-to-one onto another D_i element. This is so because for some j, $\sigma^j(B) = B(id)$ and hence $\sigma_M(B) \in D_i$ by Lemma 4.2.5. The following results concern the conditions for σ to produce only identity permutations in the cascade.

Definition 4.2.3: If C is a Zeiger cover for M and $\sigma_M \in \tilde{S}_M$, then $\underline{\sigma_M}$ permutes C means that, for each $B \in C$, $\sigma_M(B) = M(B,\sigma) \in C$, and the map on C so defined is one-to-one.

Clearly, $\underline{\sigma_M}$ is identity on C means that for each $B \in C$, $\sigma_M(B) = B$, but σ_M is not necessarily the identity permutation of the state subset B.

Again, we let $C_1,\ldots,C_n$ be a sequence of Zeiger covers for M with initial similarity classes $D_1,\ldots,D_{n-1}$ and corresponding component machines $L_1,\ldots,L_n$.

Lemma 4.2.7: If σ_M is identity on C_u, σ_M is identity on C_i for each $i \leq u$.

Proof: It is sufficient to show that σ_M is identity on C_{u-1}. If $B \in D_{u-1}$ then B is the union of the subsets of Q_M which replace B to obtain C_u. But each of these subsets is permuted by σ_M by assumption; hence each is the union of minimal σ-permuted subsets of Q_M, and therefore so is B. So $\sigma_M(B) = B$ by Lemma 4.2.1. Hence σ_M is identity on D_{u-1}. Since σ_M is also identity on $C_{u-1} \cap C_u$ by assumption, the lemma follows.

Q.E.D.

Theorem 4.2.6: If $\sigma_M(Q_M) = Q_M$, then for each $u \leq n$ σ_M is identity on C_u if and only if the permutations due to σ of component machines L_1 through L_u are identities.

Proof: For $u = 1$ the theorem follows directly by construction of L_1.

THE

Assume $u > 1$ and suppose σ_M is identity on C_u. Let $C_{u-1} = \{S_1,\ldots,S_{m_{u-1}}\}$ and $D_{u-1} = \{S_1,\ldots,S_{k_{u-1}}\}$ for $k_{u-1} \le m_{u-1}$, and let S_1 be the reference element. Let $R(S_i) = \{S_{i1},\ldots,S_{ir_{u-1}}\}$ for each $i \le k_{u-1}$ and assume the elements of $R(S_i)$ are labelled so that $v_{S_1}^{S_i}(S_{1j}) = S_{ij}$ for each $i \le k_{u-1}$ and each $j \le r_{u-1}$.

By Lemma 4.2.7 σ_M is identity on C_{u-1}. Hence $\sigma_M(S_{ij}) = S_{ij}$. Consider any transition of the form $\sigma_{N_u}(p,r) = (s,t)$, where $Z_{N_{u-1}}(p) = S_i \in D_{u-1}$ and $Z_{N_{u-1}}(s) = S_m \in D_{u-1}$. Then $\sigma_M(S_i) \subseteq S_m$. But $\sigma_M(S_i) = S_i$ by assumption, so $S_m = S_i$ since D_{u-1} is chosen among the elements of C_{u-1} of maximal cardinality. We will show $t = r$, where $r = S_{1j}$, for each $j \le r_{u-1}$. Now t is chosen so that $v_{S_i}^{S_1}\sigma_M v_{S_1}^{S_i}(S_{1j}) = t$. Hence $t = v_{S_i}^{S_1}\sigma_M(S_{ij}) = v_{S_i}^{S_1}(S_{ij}) = S_{1j} = r$, showing that the permutation of L_u due to σ is the identity. By Lemma 4.2.7 σ_M is identity on C_i for $i \le u$, so the same argument applies to L_i for each $i \le u$.

Conversely, suppose the permutations due to σ of L_1 through L_u are identities. We will use induction to show that σ_M is identity on C_u.

Clearly, σ_M is identity on C_1 by Theorem 4.1.1. Suppose σ_M is identity on C_v for $1 \le v < u$. Let $C_v = \{B_1,\ldots,B_m\}$ and $D_v = \{B_1,\ldots,B_{k_v}\}$ for $k_v \le m_v$, and let B_1 be the reference element. Let $R(B_i)= \{B_{i1},\ldots,B_{ir_v}\}$ for each $i \le k_v$ and assume the elements of $R(B_i)$ are labelled so that $v_{B_1}^{B_i}(B_{1j}) = B_{ij}$ for each $i \le k_v$ and $j \le r_v$.

Since $\sigma_M(B_i) = B_i$ for each $B_i \in C_v$, and since each permutation of L_{v+1} due to σ is the identity, Rule 4 of Theorem 4.1.2 for determining state transitions becomes $v_{B_i}^{B_1}\sigma_M v_{B_1}^{B_i}(B_{1j}) = B_{1j}$ for $i \le k_v$ and $j \le r_v$. Applying $v_{B_1}^{B_i}$ to both sides of this equality yields $\sigma_M[v_{B_1}^{B_i}(B_{1j})] = v_{B_1}^{B_i}(B_{1j})$. But any element of $R(D_v)$ can be written as

THE

$v_{B_1}^{B_i}(B_{1j})$ for some i,j, so σ_M is identity on $R(D_v)$, which is contained in C_{v+1}. Since σ_M is identity on C_v by assumption, it is identity on all of C_{v+1}.

We conclude that σ_M is identity on each cover $C_1,\ldots,C_u$.

Q.E.D.

Corollary 4.2.1: If σ_M permutes Q_M, then σ_M is the identity on Q_M if and only if every component machine in a Zeiger cascade for M has only identity permutations due to σ.

Proof: If $C_1,\ldots,C_n$ is a Zeiger cascade for M then $C_n = Q_M$, and the conclusion follows from Theorem 4.2.6.

Q.E.D.

This corollary shows that any non-trivial permutation of the state set by an input σ causes a non-trivial permutation in one or more machines in the cascade. Furthermore, if σ_M permutes Q_M one can determine which of the component machines have non-trivial permutations due to σ by observing the effect of σ_M on the Zeiger covers.

4.3. An Application to Definite Automata

Cutlip [5] pointed out that prefix automata, i.e., automata which recognize events of the form $\Sigma^* Z$ where $Z \subseteq \Sigma^k$ for some positive integer k, decompose into a cascade having only resets and identity permutations. These permutations arise from the arbitrary next-state assignment of Rule 2 of Theorem 4.1.2, which we have preferred to call "don't cares." In this section we consider decomposition of the more general class of machines, definite automata. Recall from Chapter 3 that these recognize events of the form $R = P + \Sigma^* Q$ where P and Q are finite and one or both are non-empty.

Notation will be taken from both this and earlier chapters, but no confusion should arise.

Let W be a regular event and $M(W) = (Q_M, M, s_0, F)$ the machine recognizing W. Σ is the input alphabet.

Lemma 4.3.1: If no non-singleton subset of Q_M is permuted by any tape in $\{\Sigma^* - \Lambda\}$ then there is an integer m such that every tape of length $\geq$ m synchronizes M(W).

THE

Proof: If no non-singleton subset of Q_M is permuted by any tape in $\{\Sigma^* - \Lambda\}$ then S_{x^∞} is a singleton for each $x \in \{\Sigma^* - \Lambda\}$. If $M(W)$ has n states then Q_M has 2^n subsets. Let $m = 2^n$. Suppose $y \in \Sigma^*$, $\ell(y) = k \geq m$, and $y = \sigma_1 \dots \sigma_k$. For each i such that $1 \leq i \leq k$, define $A_i = (\sigma_1 \dots \sigma_i)_M(Q_M)$. Since y has length greater than the number of subsets of Q_M there are integers u,v such that $v > u$ and $A_v = A_u$. Then the tape $\sigma_{u+1} \dots \sigma_v$ maps A_u to A_u; that is, $(\sigma_{u+1} \dots \sigma_v)_M(A_u) = A_v = A_u$. Hence A_u is a singleton because $S_{(\sigma_{u+1} \dots \sigma_v)^\infty}$ is a singleton and no non-singleton subset of Q_M can be permuted by $\sigma_{u+1} \dots \sigma_v$. This shows that $A_k = y_M(Q_M)$ is also a singleton and thus y synchronizes $M(W)$. So every tape of length $\geq m = 2^n$ synchronizes $M(W)$.

Q.E.D.

Lemma 4.3.2: Suppose $R = P + \Sigma^* Q$ where P is finite and R is in canonical form. Then Q is infinite if and only if Q_M has a non-singleton subset B which is permuted by some non-null tape $x \in \Sigma^*$.

Proof: Since R is in canonical form, no tape in Q is a proper suffix of another tape in Q.

Suppose $B \subseteq Q_M$ is permuted by x, where B is not a singleton and $x \neq \Lambda$. For some integer k, x^k_M is the identity on B; let $y = x^k$. Let s and t be distinct states in B and let $z \in \Sigma^*$ distinguish them. Without loss of generality assume $M(s,z) \in F$ and $M(t,z) \notin F$. Choose tapes u and v such that $M(s_0,u) = s$ and $M(s_0,v) = t$. Let $R_1 = uy^*z$ and $R_2 = vy^*z$ and observe that $R_1 \subseteq R$ and $R_2 \cap R = \phi$. Since P is finite, an infinite number of tapes in R_1 must be in $\Sigma^* Q$. If $T = R_1 \cap \Sigma^* Q$, denote T by $T = \{uy^{i_1}z, uy^{i_2}z, uy^{i_3}z, \dots\}$, where each i_j is a non-negative integer and $i_j > i_{j-1}$ for $j = 1,2,3,\dots$. Since $R_2 \cap R = \phi$, no suffixes of tapes in R_2 are in Q, so in particular no suffix of $y^i z$ is in Q for $i = 0,1,2,\dots$ But every tape in T has a suffix in Q. Hence there are suffixes u_j of u for $j = 1,2,3,\dots$ such that if u_j is not a suffix of y^m for any j,m, then the following tapes are in Q: $u_1y^{i_1}z$, $u_2y^{i_2}z$, $u_3y^{i_3}z, \dots$. But u_j cannot be a suffix of y^m, for if j is the

THE

smallest index such that u_j has this property, then $u_j y^{i_j} z \in Q$, which implies that $y^m y^{i_j} z \in \Sigma^* Q$, and the fact that no suffix of $y^i z$ is in Q for $i = 0,1,2,\ldots$ is contradicted. Hence Q has an infinite number of tapes.

Conversely, assume that no non-singleton subset of Q_M is permuted by any tape in $\{\Sigma^* - \Lambda\}$. By Lemma 4.3.1 tapes of length $\geq m = 2^n$ synchronize $M(R)$, where $M(R)$ has n states. Suppose the longest tape in P has length r, and suppose y is any tape in $\Sigma^* Q$ such that $\ell(y) > \max\{m,r\} + 1$. Let $y = wx$, where x is a suffix of y of length $\max\{m,r\} + 1$. Now y synchronizes $M(R)$ to some state $t \in F$. Let $s = M(s_0,w)$. Since $\ell(x) > m$, x synchronizes $M(R)$, and since $M(s,x) = t$, x synchronizes to state t. This means $M(s_0,x) = t$, so that $x \in R$. But $\ell(x) > r$, so $x \notin P$. Hence $x \in \Sigma^* Q$ and so has a suffix in Q. Then y has a suffix in Q, so $y \notin Q$ since R is in canonical form. Thus no tape in $\Sigma^* Q$ of length greater than $\max\{m,r\} + 1$ is in Q, so Q is a finite set.

Q.E.D.

The main theorem can now be stated.

<u>Theorem 4.3.1</u>: R is definite if and only if no Zeiger cascade for $M(R)$ has permutations.

<u>Proof</u>: Assume R is definite and in canonical form. Then $R = P + \Sigma^* Q$ and Q is finite, so by Lemma 4.3.2 no non-singleton subset of Q_M is permuted by any non-null tape. That is, S_{x^∞} is a singleton for all $x \in \{\Sigma^* - \Lambda\}$, and by Theorem 4.2.5 a Zeiger cascade for $M(R)$ has no permutations.

Conversely, if no Zeiger cascade for $M(R)$ has permutations, then by Theorem 4.2.5 S_{x^∞} is a singleton for all $x \in \{\Sigma^* - \Lambda\}$. By Lemma 4.3.1 every tape of length $\geq m = 2^n$ synchronizes $M(R)$, where $M(R)$ has n states.

If $u \in R$ and u synchronizes $M(R)$, then $\Sigma^* u \subseteq R$. Let Q be the set of all tapes in R of length $\geq m$. Then $\Sigma^* Q \subseteq R$. Letting P be the set of all tapes in R of length less than m, we see that $R = P + \Sigma^* Q$ where P is finite.

Let $R = P' + \Sigma^* Q'$ be the canonical form for R. P' is still

THE

finite, and since Q_M has no non-singleton subsets permuted by tapes in $\{\Sigma^* - \Lambda\}$, Q' is finite by Lemma 4.3.2. Hence R is definite.

Q.E.D.

THE

5. SUMMARY AND CONCLUSIONS

We summarize in this chapter some of the more important results of the study and suggest areas for further research.

Relations between certain structural aspects of automata and their accepted events are the object of the preceding investigation. The structural aspects of an automaton are expressed in terms of the state graph, and event structure is given by the properties of the regular expression describing the event.

In Chapter 2 general repetitive machines are defined. This class of automata consists of machines that have a path from every final state to the start state, and contains as a proper subclass the strongly connected machines. A class of events, called general repetitive events, is defined in terms of certain "factorization" properties of the associated regular expressions. The main result of the chapter is Theorem 2.2.1, which shows a one-to-one onto correspondence between general repetitive machines and general repetitive events. These "factorization" properties of the regular expressions for GREs result from the "loop" structure of the paths linking start and final states in the GRM.

The concept of synchronization, which has been investigated by other authors, [11, 13], is introduced in Chapter 2. It is shown that all non-strongly-connected GRMs are synchronizable.

Definite and ultimate-definite GRMs are the topic of Chapter 3, where it is demonstrated that these are exactly the strongly connected machines of each class. A number of properties of ultimate-definite GRMs are shown to be equivalent, among them the properties of general repetitiveness, strong connectedness, synchronizability to the start state, and, in the case of events of the form $\Sigma^{*}P^{t}$ with P^{t} finite, the absence from P^{t} of tapes beginning with certain symbols. These equivalent properties depend on the fact that all ultimate-definite automata are synchronizable (Theorem 3.2.3). Chapter 3 concludes with several results concerning the canonical representations of the union and intersection of ultimate-definite events, and with closure properties of certain subclasses of the events considered in this study.

THE

In Chapter 4, the structure question is pursued from the viewpoint of a machine's cascade decomposition. The cascade decomposition procedure of Zeiger [16] is presented first, so that all the known corrections to his original version might appear in one place. An input to a component machine of a Zeiger cascade either permutes the states or resets to one state, and Chapter 4 focuses on the properties of the machine being decomposed which cause such permutations or resets. It is shown that certain subsets of a machine that are permuted under some input appear as cover elements somewhere in a Zeiger cascade, and permutations of component machines appear when these subsets are not singletons. From this result, necessary and sufficient conditions are obtained for a machine to have a Zeiger cascade with no permutations. For the special case where a symbol σ permutes the whole state set of a machine, necessary and sufficient conditions are obtained for σ to produce only identity permutations in the machine's Zeiger cascade.

Perhaps the most interesting result of Chapter 4 is the application to definite automata, where it is proved that an event R is definite if and only if no Zeiger cascade for $M(R)$ has permutations.

Several topics that warrant further research arise from this study. It would be interesting, for example, to study the general repetitive machines associated with other classes of events, such as star events (that is, events of the form P^*), and to characterize the way these machines decompose via the Zeiger method. Also, those machines which are restartable (synchronizable to the start state) should be investigated since the restartability property is useful in such practical situations as error correcting. Finally, it would be enlightening to attack the problems investigated in Chapter 4 using the semigroup approach of Krohn and Rhodes.

THE

LIST OF REFERENCES

THE

LIST OF REFERENCES

1. BRZOZOWSKI, J.A. Canonical regular expressions and minimal state graphs for definite events. Proc. Symp. on Mathematical Theory of Automata, Polytechnic Press, Brooklyn, New York (1962), 529-561.

2. ________. Derivatives of regular expressions. J. ACM 11 (Oct. 1964), 481-494.

3. ________. Roots of star events. J. ACM 14 (July 1967), 466-477.

4. CUTLIP, W.F. Amendments to Zeiger's cascade synthesis of finite-state machines. To appear in Inf. Contr.

5. ________. On the cascade decomposition of prefix automata. IEEE Trans. on Computers C-17 (Jan. 1968), 94-95.

6. ________. Synchronization and decomposition of finite automata. Ph.D. thesis, Department of Mathematics, Michigan State University, East Lansing, Michigan, 1968.

7. HARTMANIS, J. and Stearns, R.E. Algebraic Structure Theory of Sequential Machines. Prentice-Hall, Englewood Cliffs, New Jersey, 1966.

8. KLEENE, S.C. Representation of events in nerve nets and finite automata. Automata Studies, Ann. Math. Studies No. 34, C.E. Shannon and J. McCarthy (Eds.), Princeton U. Press, Princeton, N.J. (1956), 3-41.

9. KROHN, K.B. and Rhodes, J.L. Algebraic theory of machines. In Proc. Symp. on Mathematical Theory of Automata, Polytechnic Press, Brooklyn, New York (1962), 341-384.

10. ________. Algebraic theory of machines: I. Prime decomposition theorem for finite semigroups and machines. Trans. Am. Math. Soc. 116 (1965), 450-464.

11. LIU, C.L. Some memory aspects of finite automata. Sc.D. dissertation, Massachusetts Inst. Tech., Cambridge; Research Lab. of Electronics, Cambridge, Tech. Rpt. No. 411 (May 31, 1963).

12. MOORE, E.F. Gedanken-experiments on sequential machines. Automata Studies, Ann. Math. Studies No. 34, C.E. Shannon and J. McCarthy (Eds.), Princeton U. Press, Princeton, N.J. (1956), 129-153.

THE

13. NEUMAN, P.G. Error-limiting coding using information-lossless sequential machines. IEEE Trans. on Inf. Theory IT-10 (April 1964), 108-115.

14. PAZ, A. and Peleg, B. Ultimate-definite and symmetric-definite events and automata. J. ACM 12 (July 1965), 399-410.

15. PERLES, M., Rabin, M.O. and Shamir, E. The theory of definite automata. IEEE Trans. on Electronic Computers EC-12 (June 1963), 233-243.

16. ZEIGER, H.P. Cascade synthesis of finite-state machines. In Proc. of the Sixth Annual Conf. on Switching Theory and Automata, Institute of Electrical and Electronics Engineers, New York, 1965.

17. ________. Cascade synthesis of finite-state machines. Inf. Contr. 10 (1967), 419-433.

18. Erratum. Inf. Contr. 11 (1967), 471.

THE

APPENDIX

THE

APPENDIX

Completion of proof of Theorem 4.1.2.

To show that N tells where M is in C', consider transitions $\sigma_N(p,r) = (s,t)$ of Q_N, where $\sigma \in I_N$, $Z_K(p) = P \in C$, $Z_K(s) = S \in C$, and r and t are states of L and hence subsets of U. By hypothesis K tells where M is in C, so $\sigma_M[Z_K(p)] \subseteq Z_K(s)$. We must show that $\sigma_M[Z_N(p,r)] \subseteq Z_N(s,t)$.

<u>Case 1</u>: If $P \notin D$ and $S \notin D$ then $Z_N(p,r) = Z_K(p)$ and $Z_N(s,t) = Z_N(s)$. Hence $\sigma_M[Z_N(p,r)] \subseteq Z_N(s,t)$.

<u>Case 2</u>: If $P \notin D$ and $S \in D$ then $Z_N(p,r) = Z_K(p) = P$ and $Z_N(s,t) = v_U^S(t)$, and by Rule 3 t is chosen so that $v_S^U\sigma_M(P) \subseteq t$. Applying the transformation v_U^S and observing that $v_U^S v_S^U$ is the identity map on S, we obtain $v_U^S v_S^U \sigma_M(P) = \sigma_M(P) \subseteq v_U^S(t)$, or $\sigma_M[Z_N(p,r)] \subseteq Z_N(s,t)$.

<u>Case 3</u>: If $P \in D$ and $S \notin D$ then $Z_N(p,r) = v_U^P(r)$ and $Z_N(s,t) = Z_K(s) = S$. By hypothesis $\sigma_M(P) \subseteq S$, and since $v_U^P(r) \subseteq P$ the result $\sigma_M[Z_N(p,r)] = \sigma_M[v_U^P(r)] \subseteq \sigma_M(P) \subseteq S = Z_N(s,t)$ is obtained.

<u>Case 4</u>: If $P \in D$ and $S \in D$ then $Z_N(p,r) = v_U^P(r)$ and $Z_N(s,t) = v_U^S(t)$, and by Rule 4 t is chosen so that $v_S^U\sigma_M v_U^P(r) \subseteq t$. Thus $\sigma_M[Z_N(p,r)] = \sigma_M[v_U^P(r)] = v_U^S v_S^U[\sigma_M v_U^P(r)] \subseteq v_U^S(t) = Z_N(s,t)$.

This completes the proof that N tells where M is in C'.

THE