UNCLASSIFIED

AD 427880

DEFENSE DOCUMENTATION CENTER
FOR
SCIENTIFIC AND TECHNICAL INFORMATION
CAMERON STATION, ALEXANDRIA, VIRGINIA

UNCLASSIFIED
NOTICE: When government or other drawings, specifications or other data are used for any purpose other than in connection with a definitely related government procurement operation, the U. S. Government thereby incurs no responsibility, nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use or sell any patented invention that may in any way be related thereto.
The Decomposition of Finite State Machines

THOMAS V. GRIFFITHS
Requests for additional copies by Agencies of the Department of Defense, their contractors, and other government agencies should be directed to the:

Defense Documentation Center
Cameron Station
Alexandria, Virginia

Department of Defense contractors must be established for DDC services, or have their 'need-to-know' certified by the cognizant military agency of their project or contract.

All other persons and organization should apply to the:

U.S. DEPARTMENT OF COMMERCE
OFFICE OF TECHNICAL SERVICES,
WASHINGTON 25, D.C.
Research Report

The Decomposition of Finite State Machines

THOMAS V. GRIFFITHS

This material was originally submitted in partial fulfillment of the requirements for the degree of Master of Science in Electrical Engineering at Massachusetts Institute of Technology, June 1963.

DATA SCIENCES LABORATORY PROJECT 4641
AIR FORCE CAMBRIDGE RESEARCH LABORATORIES, OFFICE OF AEROSPACE RESEARCH, UNITED STATES AIR FORCE, L.G. HANSCOM FIELD, MASS.
Abstract

This work is concerned with decomposition of a finite state machine into the Cartesian product of two smaller finite state machines.

The first half of the work is expository. A notation is developed to treat the problem, and principal results of Hartmanis and Yoeli are given with some extensions by the author.

In the second half of the work, an addition operation on finite state machines is defined and it is shown that the product operation distributes over the addition operation (Theorem 3).

It is shown that the output-free finite state machine is equal to the sum of a set of single-input, output-free finite state machines.

In Section 5, some of the properties of transformation finite state machines, a special case of single-input, output-free finite state machines, are discussed. It is shown that the transformation finite state machine may be modeled by a transformation on a finite set. Some theorems are proved relating the structure of two transformation finite state machines to the structure of their product.

Generating functions for transformation finite state machines are introduced, and it is shown how these may be used in obtaining the decomposition of a transformation finite state machine as the product of two smaller transformation finite state machines if such a decomposition exists.
1. INTRODUCTION  1
2. PRELIMINARIES  1
   2.1 Conventions  1
   2.2 Functions  3
   2.3 Operations  5
   2.4 Binary Relations  8
   2.5 Notations Frequently Used  10
3. DEFINITION OF THE FINITE STATE MACHINE  12
   3.1 Concatenations  12
   3.2 Finite State Machines  15
   3.3 Special Classes of Finite State Machines  18
4. THE COMPOSITION OF FINITE STATE MACHINES  21
   4.1 Some General Remarks on Functions  21
   4.2 Machine Homomorphism, Isomorphism, and Inclusion  25
   4.3 The Product Machine  31
   4.4 Some Theorems on the Decomposability of a Finite State Machine  38
   4.5 The Sum Finite State Machine  44
   4.6 Output Free Machines  48
5. THE DECOMPOSITION OF THE SINGLE INPUT, OUTPUT FREE
   FINITE STATE MACHINE  53
   5.1 The Transformation Finite State Machine  53
   5.2 The A and A^{-1} Transformation Finite State Machines  60
   5.3 Non-Subtractable Transformation Finite State Machines  72
   5.4 Generating Functions  79
   5.5 Decomposition of a T.F.M.  91
   5.6 Multiple Generating Functions  94
6. CONCLUSION  96
ACKNOWLEDGMENTS  99
REFERENCES  101
BIBLIOGRAPHY  101
<table>
<thead>
<tr>
<th>Illustrations</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Figure</strong></td>
</tr>
<tr>
<td>1. An Example of an Operation Table</td>
</tr>
<tr>
<td>2. A Matrix Representation of a Mod 3 Adder, and a State Diagram of the Same Device</td>
</tr>
<tr>
<td>3. The F. S. M. can be Represented as the Cascade of an Output-Free F. S. M. and a Serial Encoder</td>
</tr>
<tr>
<td>4. $M_1$ is a Mod 4 Adder and $M_2$ is a Mod 2 Adder</td>
</tr>
<tr>
<td>5. $F_1, F_2$ and $F_3$ are S. F. M.'s. $F_2 \geq F_1$, but $F_1 \not\leq F_2$. $F_2 \leq F_3$, but $F_3 \not&gt; F_2$. $F_3 \geq F_1$, and $F_1 \leq F_3$.</td>
</tr>
<tr>
<td>6. $M_1$ is a Mod 3 Adder, $M_2$ is a Mod 2 Adder, and $M_3$ is a Mod 6 Adder. $M_3 \cong M_1 \oplus M_2$.</td>
</tr>
<tr>
<td>7. $M_1, M_2$ and $M_3$ are S. F. M.'s. $M_2 \otimes M_1, \cong M_3 \cong M_2 \otimes M_2$ but $M_1 \not\cong M_2$.</td>
</tr>
<tr>
<td>8. $M_1, M_2, M_3, M_4$ and $M_5$ are O. F. M.'s. $M_1 = M_2 + M_3 = M_4 + M_5$</td>
</tr>
<tr>
<td>9. $F$ is a T. F. M. $(S, \Lambda)$, where $\Lambda$ is defined by the State Diagram of $F$. Shown also are $F_\infty, \Lambda F, \Lambda^{-1}F_\infty, \Lambda^{-2}F_\infty$</td>
</tr>
<tr>
<td>10. A N. S. T. F. M. whose $(n, \Lambda^{-1})$, d. g. f. is $A_0x^0 + A_1x^1 + \ldots + A_nx^n$</td>
</tr>
<tr>
<td>11. Example of the Composition of Two T. F. M.'s</td>
</tr>
</tbody>
</table>
The Decomposition of Finite State Machines

1. INTRODUCTION

A finite state machine is a physical device which has discrete sets of inputs and outputs, a bounded amount of storage, a sequential mode of operation, and a deterministic behavior (as opposed to, say, a probabilistic behavior). Later, a precise mathematical model for the finite state machine is given. Some interesting examples of finite state machines are combination locks, switching circuits, and digital computers with a bounded amount of storage. More examples may be found in Gill.¹

This report discusses the decomposition of a finite state machine into two or more smaller finite state machines which operate in parallel, and whose combined behavior is the same as the behavior of the decomposed finite state machine. One of the chief analytical tools in this report is the single input, output free, finite state machine, a device already discussed by Yoeli² in connection with the decomposition problem. The results obtained here extend Yoeli's results as well as some of the results of Hartmanis.³

2. PRELIMINARIES

2.1 Conventions

This section presents notation and conventions which are basic for the under-

Received for publication 23 August 1963.
standing of the following sections. Other less basic ideas are presented as needed in the development of this work.

The numbers appearing in this report are non-negative integers unless otherwise stated.

The phrase, "such that", is used quite frequently and is abbreviated "s.t."

The symbolism

\[ \text{Ex (} R(x) \text{)} \]

means: "There exists an \( x \) s.t. \( R(x) \)." For example, "Ex (\( x < 3 \))" means: "There exists an \( x \) s.t. \( x < 3 \)."

The symbolism

\[ \text{Ex} (\neg R(x)) \]

means that it is not the case that Ex (\( R(x) \)).

Objects to be specified in some order, say \( a_1, a_2, \ldots \), are denoted by

\[ (a_1, a_2, \ldots) \]

The objects do not have to be distinct. A finite ordered arrangement of objects is called an \( n \)-tuple (or sometimes a pair, or triple, or etc.), where \( n \) is the number of elements in the ordered arrangement.

When

\[ (a_1, a_2, \ldots, a_n) = (b_1, b_2, \ldots, b_m) \]

is written, the meaning is

\[ n = m \text{ and } a_i = b_i \quad 1 \leq i \leq n \]

Two methods are used to denote sets. The set consisting of the elements \( a_1, a_2, \ldots \) is denoted by

\[ \{ a_1, a_2, \ldots \} \]

The set

\[ \{ x \mid R(x) \} \]

is the set of all \( x \) s.t. \( R(x) \). For example, \( \{ x \mid x < 3 \} \) is the set of all numbers less than 3.
If $A$ is a set, the set of all subsets of $A$ (sometimes called the power set of $A$) is denoted by

$2^A$

If $A$ is finite, the number of elements in $A$ is denoted by

$\# (A)$

Three sets are given the following symbols:

$\Phi = \text{the empty set},$

$L = \{0, 1\},$

and

$Z = \{0, 1, 2, \ldots \}.$

2.2 Functions

If $A$ and $B$ are sets,

$f : A \rightarrow B$

means "$f$ is a function from $A$ into $B."$ The set $A$ is called the domain of $f,$ and the set $B$ is called the co-domain of $f.$ A function from $A$ into $B$ assigns to each element of $A,$ an element of $B.$ If $a$ is an element of $A,$

$f(a)$

denotes the element of $B$ assigned to $a$ by $f.$ The element $a$ is called the argument of $f(a).$ The set

$\{ f(a) \mid a \text{ is in } A \}$

is called the range of $A.$ When $A = \{a_1, a_2, \ldots, a_n\}$ is finite, $f$ may be defined by the notation

$\begin{pmatrix}
  a_1 & a_2 & \cdots & a_n \\
  b_1 & b_2 & \cdots & b_n
\end{pmatrix}$
where

\[ b_i = f(a_i), \quad 1 \leq i \leq n \]

If \( f : A \to B \) and \( C \) is a subset of \( A \), the function

\[ f|_C : C \to B, \]

called the restriction of \( f \) to \( C \), is defined by

\[ f|_C(c) = f(c) \]

for all \( c \) in \( C \).

A function \( f : A \to B \) may be said to have certain properties. Three of the most common are:

1. Onto is said to hold if the range of \( f \) is all of \( B \).
2. One-one is said to hold if

\[ f(a_1) = f(a_2) \]

implies

\[ a_1 = a_2 \]

for all \( a_1 \) and \( a_2 \) in \( A \).

3. Invertibility is said to hold if \( f \) is both onto and one-one.

Consider \( f : A \to B \). If \( f \) is invertible, there is a unique function \( f^{-1} : B \to A \), called the inverse of \( f \), which satisfies

\[ f^{-1}(f(a)) = a \]

for all \( a \) in \( A \). For any \( b \) in \( B \), the onto property of \( f \) guarantees that

\[ \exists a \ (a \text{ is in } A \text{ and } b = f(a)) \],

while the one-one property of \( f \) guarantees that there is at most one such \( a \). It is this unique \( a \) that is assigned to \( b \) by \( f^{-1} \). This shows that \( f^{-1} \) is well defined. The functions \( f \) and \( f^{-1} \) have the additional properties that \( f^{-1} \) is invertible and

\[ (f^{-1})^{-1} = f. \]
First, it is obvious that
\[ f(f^{-1}(b)) = b \]
for all \( b \) in \( B \), since \( f^{-1}(b) \) is defined to be the \( a \) s.t. \( f(a) = b \). Second, \( f^{-1}(f(a)) = a \) for all \( a \) in \( A \) implies \( f^{-1} \) is onto, while \( f(f^{-1}(b)) = b \) for all \( b \) in \( B \) implies \( f^{-1} \) is one-one, otherwise \( f \) wouldn’t be a well defined function. Hence it follows that \( f^{-1} \) is invertible and \( (f^{-1})^{-1} = f \). When \( f \) is invertible, the sets \( A \) and \( B \) are said to be in one to one correspondence.

If
\[ f : A \to A, \]
\( f \) is called a transformation on the set \( A \). If \( f \) is invertible, it is called a permutation on \( A \). The symbol \( I_A \) is used to denote the identity permutation on \( A \); that is, the permutation on \( A \) is defined by
\[ I_A(a) = a \]
for all \( a \) in \( A \).

2.3 Operations

Suppose \( A \) is a set. An operation on \( A \) is a function \( f \) from the set of all pairs of elements in \( A \) into \( A \). If \( a_1 \) and \( a_2 \) are in \( A \), then \( f(a_1, a_2) \) is designated by
\[ a_1 \ast a_2 \]
where \( \ast \) is the operation symbol. An example of an operation is addition on \( Z \) (the set of all non-negative integers), where the \( + \) is the operation symbol. There may, of course, be more than one operation on \( A \). A convenient notation for a set of operations, whose symbols are \( \ast_1, \ast_2, \ldots \), on the set \( A \) is
\[(A, \ast_1, \ast_2, \ldots).\]

When \( A \) is finite, the operations on \( A \) may be specified by operation tables. For example, Figure 1 shows the operation table for \((A, \ast)\), where \( A = \{a_1, a_2, \ldots, a_n\} \).

A set and its operations may be said to have certain properties. A few common ones are given below. Consider \((A, \ast)\):
Figure 1. An Example of an Operation Table

1. Associativity is said to hold if

\[ a_1 \ast (a_2 \ast a_3) = (a_1 \ast a_2) \ast a_3 \]

for all \( a_1, a_2, \) and \( a_3 \) in \( A \).

2. Commutativity is said to hold if

\[ a_1 \ast a_2 = a_2 \ast a_1 \]

for all \( a_1 \) and \( a_2 \) in \( A \).

3. Identity is said to hold if

\[ a \ast e = e = e \ast a \quad \text{for all} \ a \ in \ A \]

It is easily proved that the identity element is unique. If \( e \) and \( e' \) are both identity elements, then

\[ e \ast e' = e = e' \]

4. Inverse is defined only when identity holds and is said to hold if for each \( a \) in \( A \)

\[ a \ast a^{-1} = e = a^{-1} \ast a \]
(5) Cancellation is said to hold if
\[ a_1 \hat{*} a_2 = a_1 \hat{*} a_3 \]
implies
\[ a_2 = a_3 \]
for all \( a_1, a_2, \) and \( a_3 \) in \( A \).
Consider \( (A, \hat{+}_1, \hat{+}_2) \):

(6) Distributivity of \( \hat{+}_1 \) over \( \hat{+}_2 \) is said to hold if
\[ a_1 \hat{+}_1 (a_2 \hat{+}_2 a_3) = (a_1 \hat{+}_1 a_2) \hat{+}_2 (a_1 \hat{+}_1 a_3) \]
and
\[ (a_2 \hat{+}_2 a_3) \hat{+}_1 a_1 = (a_2 \hat{+}_1 a_1) \hat{+}_2 (a_3 \hat{+}_1 a_1) \]
for all \( a_1, a_2, \) and \( a_3 \) in \( A \).

If \( (A, \hat{+}) \) satisfies (1), \( (A, \hat{+}) \) is called a semi-group; if \( (A, \hat{+}) \) satisfies (1) and (3), \( (A, \hat{+}) \) is called a monoid; and if \( (A, \hat{+}) \) satisfies (1), (3), and (4), \( (A, \hat{+}) \) is called a group.

If \( (A, \hat{+}) \) is a group and \( a \) is in \( A \), then the inverse of \( a \) is unique and is written \( a^{-1} \). To show the uniqueness of the inverse of \( a \), suppose that \( a_1 \) and \( a_2 \) are both inverses of \( a \). Then
\[ a_1 = a_1 \hat{+} (a \hat{+} a_2) \]
\[ (a_1 \hat{+} a) \hat{+} a_2 = a_2 \]

Let \( (A, \hat{+}) \) be a set and an operation on that set. Suppose \( B \) is a subset of \( A \), and \( b_1 \hat{+} b_2 \) is in \( B \) for all \( b_1 \) and \( b_2 \) in \( B \); then \( (B, \hat{+}) \) is a subgroup (or subsemi-group or submonoid) of \( (A, \hat{+}) \) if \( (B, \hat{+}) \) satisfies the conditions for a group (or semi-group or monoid). If \( (A, \hat{+}) \) is a group, and \( (B, \hat{+}) \) is a subsemi-group (or submonoid) of \( (A, \hat{+}) \), then \( (B, \hat{+}) \) is cancellative. Suppose \( b_1, b_2, \) and \( b_3 \) are all in \( B \). Then
\[ b_1 \hat{+} b_2 = b_1 \hat{+} b_3 \]
implies
\[ b_1^{-1} \cdot b_1 \cdot b_2 = b_1^{-1} \cdot b_1 \cdot b_3 \]
and
\[ b_2 = b_3. \]
since \( b_1^{-1} \) is guaranteed to exist in \( A \). In similar fashion, cancellation on the right can be shown to hold, and it follows that \((B, \cdot)\) must be cancellative to be a subsemi-group (or submonoid) of \((A, \cdot)\).

2.4 Binary Relations

If \( A \) is a set, a binary relation on \( A \) is a function from the set of pairs of elements of \( A \) into \( L \). If \( R \) is a binary relation on \( A \), and \( a_1 \) and \( a_2 \) are members of \( A \), then
\[ R(a_1, a_2) = 0 \text{ or } 1. \]

If \( R \) is a binary relation, the following notation is common. Write
\[ aRb \text{ if } R(a, b) = 1 \]
and
\[ a\not{\sim}b \text{ if } R(a, b) = 0. \]

Examples of binary relations on \( \mathbb{Z} \) are \( = \) and \( \geq \).

Let \( A \) be a set and \( R \) a binary relation on \( A \); then \( R \) may be said to have certain properties on the set \( A \). Common properties are:

(1) Reflexivity is said to hold when
\[ aRa \]
for all \( a \) in \( A \).

(2) Symmetry is said to hold when
\[ a_1Ra_2 \]
implies
for all \( a_1 \text{ and } a_2 \) in \( A \).

(3) Anti-symmetry is said to hold when

\[ a_1 R a_2 \text{ and } a_2 R a_1 \]

together imply

\[ a_1 = a_2 \]

for every \( a_1 \text{ and } a_2 \) in \( A \).

(4) Transitivity is said to hold when

\[ a_1 R a_2 \text{ and } a_2 R a_3 \]

together imply

\[ a_1 R a_3 \]

for all \( a_1, a_2, \text{ and } a_3 \) in \( A \).

If \( R \) satisfies (1), (2), and (4), then \( R \) is called an equivalence relation on \( A \); and if \( R \) satisfies (1), (3), and (4), then \( R \) is called a partial order relation on \( A \).

A set \( A \) is said to be partitioned into the subsets \( A_1, A_2, \ldots, A_n \) if each element of \( A \) occurs in some \( A_i \) and only in that \( A_i \). Suppose \( R \) is an equivalence relation on \( A \) and

\[ [a_i] = \{a_j \mid a_i R a_j \} \]

then \( A \) is partitioned into

\[ \{ [a_i] \mid a_i \text{ is in } A \} \]

Reflexivity implies that for each \( a \) in \( A, a \) is in \( [a] \). Suppose \( a \) is in both \( [a_1] \text{ and } [a_j] \). Then

\[ a_1 R a \text{ and } a_j R a. \]
By symmetry and transitivity, it follows that

\[ a_i R a_j, \]

and by transitivity again it follows that all members of \([a_i]\) are members of \([a_j]\), and vice versa. This shows that \([a_i]\) and \([a_j]\) are identical, and hence it follows that no element of \(A\) is a member of more than one of the members of \(\{ [a_i] | a_i \text{ is in } A \}\).

2.5 Notations Frequently Used

Some common operations and binary relations which will be used frequently are enumerated below. Suppose \(A\) and \(B\) are sets:

1. The expression
   \[ a \in A \]
   means \(a\) is a member of \(A\).

2. The expression
   \[ B \subseteq A \text{ or } A \supseteq B \]
   is to be read "\(B\) is a subset of \(A\)," this means that every element of \(B\) is in \(A\).

3. The expression
   \[ A = B \]
   is to be read "\(A\) equals \(B\)," this means that \(A \subseteq B\) and \(B \subseteq A\).

4. The expression
   \[ B \subset A \text{ or } A \supset B \]
   is to be read "\(B\) is a proper subset of \(A\)," this means that \(B \subseteq A\) but \(A \neq B\).

5. The set
   \[ A \cap B, \]
   \[ = \{ c | c \in A \text{ and } c \in B \}, \]
   is called the intersection of \(A\) and \(B\).
(6) The set

\[ A \cup B, \]
\[ = \{ c \mid c \in A \text{ or } c \in B \}, \]

is called the union of \( A \) and \( B \).

(7) The set

\[ A - B, \]
\[ = \{ c \mid c \in A \text{ and } c \notin B \}, \]

is called the difference of \( A \) and \( B \).

(8) The set

\[ A \otimes B, \]
\[ = \{ (a, b) \mid a \in A \text{ and } b \in B \}, \]

is called the Cartesian cross-product of \( A \) and \( B \).

The relations =, \( \geq \), \( \leq \), \( > \), and \( < \), and the operations +, -, *, and \( \div \) on the set \( \mathbb{Z} \) all have their usual meanings.

If \( p, q, \) and \( n \in \mathbb{Z} \) and \( p \) and \( q \) have the same remainder when divided by \( n \), it is common to state this fact by writing

\[ (p = q) \mod n. \]

For example, \( (7 = 4) \mod 3 \). If

\[ Z_n = \{ 0, 1, \ldots, n-1 \}, \]

then the operations \( +_n \) and \( \cdot_n \) are defined on \( Z_n \) by

\[ x +_n y = z \text{ s.t. } z \in Z_n \text{ and } (z = x + y) \mod n, \]

and

\[ x \cdot_n y = z \text{ s.t. } z \in Z_n \text{ and } (z = x \cdot y) \mod n, \]

for all \( x \) and \( y \) in \( Z_n \).
3. DEFINITION OF THE FINITE STATE MACHINE

3.1 Concatenations

DEFINITION 1.

(1) The concatenation of the element $x$ with the element $y$ is the element $xy$.

(2) The $n$-tuple $(a_1, a_2, \ldots, a_n)$ concatenated with the $n$-tuple $(b_1, b_2, \ldots, b_n)$ is the $n$-tuple $(a_1b_1, a_2b_2, \ldots, a_nb_n)$.

(3) The set $\{a_1, a_2, \ldots, a_m\}$ concatenated with the set $\{b_1, b_2, \ldots, b_n\}$ is the set $\{a_ib_j \mid 1 \leq i \leq m \text{ and } 1 \leq j \leq n\}$.

DEFINITION 2.

Suppose $A = \{a_1, a_2, \ldots, a_n\}$ is a finite set; then a string of elements of $A$ is defined by:

(1) $a_i$ is a string if $a_i \in A$, and
(2) the concatenation of two strings is a string.

The set of all strings of elements of $A$ is denoted by $\widetilde{A}$.

In Definition 2, concatenation may be viewed as an operation on $\widetilde{A}$. The fact that

$$((a_1a_2)a_3) = (a_1(a_2a_3)) = a_1a_2a_3$$

(which holds if $a_1$, $a_2$, and $a_3$ are all elements, $n$-tuples, or sets) shows that concatenation is associative. As a result of concatenation's associativity, strings in $\widetilde{A}$ can be specified without grouping symbols.

DEFINITION 3.

Suppose $A$ is a finite set and $a_1a_2\ldots a_n$ is a string in $\widetilde{A}$, then $a_1a_2\ldots a_n$ is uniquely decodeable if
If all strings in \( \tilde{A} \) are uniquely decodeable, \( A \) is called an alphabet, and its members are called letters.

If \( A \) is an alphabet, the above definition implies that any string in \( \tilde{A} \) can be made up in only one way with letters from \( A \). The question of "whether a set \( A \) is or is not an alphabet" is important when the members of \( A \) are encoded as concatenations of members of some other set; for example, binary encoding.

**EXAMPLE 1.**

The set

\[
\{0, 1, \ldots, 9\}
\]

is an alphabet, but the set

\[
\{0, 1, \ldots, 10\}
\]

is not because, for example

\[
101 \neq (1)(0)(1) \text{ or } (10)(1).
\]

**LEMMA 1.**

If \( A \) and \( B \) are both alphabets, then:

1. \( C \) is an alphabet, if \( C \subseteq A \).
2. \( A \otimes B \) is an alphabet.
3. \( 2^A - \emptyset \) is an alphabet.

Proof:

1. Trivial.
2. Suppose

\[
a = a_1 a_2 \ldots a_n \text{ and } a' = a'_1 a'_2 \ldots a'_m \in \tilde{A}
\]

\[
b = b_1 b_2 \ldots b_n \text{ and } b' = b'_1 b'_2 \ldots b'_m \in \tilde{B}
\]

and

\[
c = (a_1, b_1) (a_2, b_2) \ldots (a_n, b_n) \text{ and }
\]

\[
c' = (a'_1, b'_1) (a'_2, b'_2) \ldots (a'_m, b'_m) \in A \otimes B.
\]
Suppose
\[ c = c' \].

Then by Definition 1(2)
\[ (a, b) = (a', b') \].

Thus
\[ a = a' \text{ and } b = b' \]

and by Definition 3
\[ n = m, \; a_i = a'_i \text{, and } b_i = b'_i, \quad 1 \leq i \leq n \]

This also shows that
\[ (a_i, b_i) = (a'_i, b'_i), \quad 1 \leq i \leq n \]

and hence that \( A \odot B \) is an alphabet since \( c \) and \( c' \) can be picked to be any two strings from \( A \odot B \).

(3) Suppose
\[ A_i \subseteq A \text{ and } A_i \neq \emptyset, \quad 1 \leq i \leq n \]

and
\[ A'_j \subseteq A \text{ and } A'_j \neq \emptyset, \quad 1 \leq j \leq m \]

Then
\[ C = A_1 A_2 \ldots A_n \text{ and } C' = A'_1 A'_2 \ldots A'_m \in 2^A - \emptyset \].

Suppose
\[ C = C' \]

and
\[ a_i \in A_i, \quad 1 \leq i \leq n \]
Then since \( C = C' \),

\[
Ea_1'a_2'\ldots a_m' (a_i' \in A_i' \text{ for } 1 \leq i \leq m)
\]

and \( a_1'a_2'\ldots a_m' = a_1a_2\ldots a_n \).

By Definition 3

\[
n = m \text{ and } a_i = a_i'.
\]

This shows

\[
a_i \in A_i',
\]

and this implies

\[
A_i \subseteq A_i'.
\]

A similar argument shows

\[
A_i' \subseteq A_i.
\]

Hence

\[
A_i = A_i'
\]

and

\[
(A_1, A_2, \ldots, A_n) = (A_1', A_2', \ldots, A_m').
\]

This shows that any string \( C \) in \( 2^A - \emptyset \) is uniquely decodeable and hence that \( 2^A - \emptyset \) is an alphabet.

3.2 Finite State Machines

DEFINITION 4.

A finite state machine (hereinafter abbreviated F.S.M.) is a quintuple of the type

\[
(S, X, Y, \Lambda, \Omega).
\]
where

(1) \( S \) is a finite alphabet, called the state alphabet.
(2) \( X \) is a finite alphabet, called the input alphabet.
(3) \( Y \) is a finite alphabet, called the output alphabet.
(4) \( \Lambda \) is a function,

\[ \Lambda : S \otimes X \rightarrow S, \]

called the next state function.
(5) \( \Omega \) is a function,

\[ \Omega : S \otimes X \rightarrow Y, \]

called the output function.

The F.S.M. may be thought of as a device capable of representing any one of several functions from \( \tilde{X} \) into \( \tilde{Y} \).

If \( s \in S \), define a function \( \Omega_s \),

\[\Omega_s : \tilde{X} \rightarrow \tilde{Y},\]

recursively by:

(1) for all \( x \in X \),

\[\Omega_s(x) = \Omega(s, x) \]

(2) and for all \( x_1 x_2 \ldots x_n \in \tilde{X} \) (\( 2 \leq n < \infty \)),

\[\Omega_s(x_1 x_2 \ldots x_n) = \Omega(s, x_1) \Omega_A(s, x_1)(x_2, x_3, \ldots, x_n).\]

Often the input and output alphabets of a F.S.M. are encoded as strings of letters from smaller alphabets. Similarly a state transition may consist of a series of 'micro-transitions'. (For example, in a digital computer, the state transition corresponding to the multiplication operation is often a series of shifts and additions.)

This is an encoding of the state transitions. In Definition 4, \( S, X, \) and \( Y \) were required to be alphabets in order that ambiguous encodings not be allowed.

For pictorially representing F.S.M.'s, two methods are common. Suppose

\[ M = (S, X, Y, \Lambda, \Omega) \]

is a F.S.M. with
\[ S = \{ s_1, s_2, \ldots, s_p \}, \]
\[ X = \{ x_1, x_2, \ldots, x_q \}, \]

and

\[ Y = \{ y_1, y_2, \ldots, y_r \}, \]

then:

(1) \( M \) may be represented by a \( p \) by \( q \) matrix whose entries are members of the set \( S \otimes Y \). If the \( ij \)-th entry in the matrix is called \( u_{ij} \) and

\[ \Lambda(s_i, x_j) = s_m \quad \text{and} \quad \Omega(s_i, x_j) = y_n, \]

then

\[ u_{ij} = (s_m, y_n). \]

(2) \( M \) may be represented by a state diagram, which is a labeled, directed graph whose nodes are the elements of \( S \). If

\[ \Lambda(s_i, x_j) = s_m \quad \text{and} \quad \Omega(s_i, x_j) = y_n, \]

then an arrow is drawn from \( s_i \) to \( s_m \). This arrow is labeled

\[ (x_j, y_n). \]

**EXAMPLE 2.**

A mod 3 adder is a F.S.M. whose input and output alphabets are both

\[ \{ 0, 1, 2 \}. \]

The inputs and outputs are related s.t. if \( y_1 \) was the last output letter and \( x \) is the next input letter, then \( y_2 \) is the output letter corresponding to \( x \), where

\[ y_2 = y_1 \mod 3. \]

Figure 2 shows both a matrix and a state diagram representation of a mod 3 adder.
### 3.3 Special Classes of Finite State Machines

**DEFINITION 5.**

An output-free F.S.M. (hereinafter abbreviated O.F.M.) is a F.S.M. of the type

$$(S, X, S \otimes X, \Lambda, I_S \otimes X).$$

Clearly the triple

$$(S, X, \Lambda)$$

contains enough information to reconstruct the quintuple description of the O.F.M. and, accordingly, a triple of the above nature is used to specify an O.F.M.

By Lemma 1(1), $S \otimes X$ in Definition 5 is an alphabet. Hence it follows that the O.F.M. is indeed a type of F.S.M.

The structure of the particular O.F.M. $(S, X, \Lambda)$ with respect to the class of all other O.F.M.'s is completely determined by the properties of $S$, $X$, and $\Lambda$. Therefore, when the relation of one O.F.M. to another is being considered, the output alphabet and output function is disregarded. In the overall operation of a F.S.M., the output is, of course, very important. Nonetheless, it is the author's opinion that in some cases a great deal of insight into the structure of F.S.M.'s can be had from consideration of the class of O.F.M.'s and one of its subclasses; in particular, that of single-input, output-free F.S.M.'s.

---

*In Hartmanis, the analysis is of output-free F.S.M.'s.*
DEFINITION 6.

A single-input O. F. M. (hereinafter abbreviated S. F. M.) is an output-free F.S. M. of the type

\[(S, \{ x \}, \Lambda)\].

Define a transformation \(\Lambda'\),

\[\Lambda': S \rightarrow S,\]

by

\[\Lambda'(s) = \Lambda(s, x)\]

for all \(s \in S\). Clearly the triple

\[(S, x, \Lambda')\]

contains enough information to reconstruct the original triple description of the S. F. M., and, accordingly, a triple of the latter nature is used to specify a S. F. M.

The S. F. M. is an O. F. M. whose input alphabet contains only one letter. If \(F = (S, x, \Lambda)\) is a S. F. M., then most of \(F\)'s important properties with respect to the class of all S. F. M.'s are determined by \(S\) and \(\Lambda\) alone, but not all of them.

It is for this reason that \(x\) is carried along as a descriptor of \(F\).

The important role of the S. F. M. in the decomposition of the O. F. M. is shown later in this report.

It is also interesting to note that the S. F. M. is a model for a digital computer with the characteristics:

1. The computer has \(p\) memory registers (including operation registers), \(m_1, m_2, \ldots, m_p\), each capable of holding a binary word of \(q\) bits.
2. Programs and data are completely stored in the \(p\) memory registers. While executing a program, the computer is not allowed to perform any sort of input operation.

Such a computer is equivalent to a S. F. M.

\[F = (S, x, \Lambda)\]

where:
(1) The set

\[ S = \{ (n_1, n_2, \ldots, n_p) \mid n_i \text{ is the binary word in register } m_i, \ 1 \leq i \leq p \}. \]

The computer's state at any given time may be completely specified by the ordered arrangement of the numbers in registers \( m_1 \) through \( m_p \); that is, by some \( s \in S \).

(2) The letter \( x \) may be interpreted as some sort of computer generated synchronizing signal.

(3) The transformation \( \Lambda \) on \( S \) may be defined by observations on the computer. If \( s \in S \), start the computer in the state represented by \( s \); if the computer's state is represented by \( s' \in S \) after the synchronizing \( x \), then

\[ \Lambda(s) = s'. \]

It is not difficult to see that

\[ \#(S) = 2^{pq} \]

which in present day large scale computers is often on the order of

\[ 2^{20} \approx 1000 \times 10^5. \]

**DEFINITION 7.**

A serial encoder (hereinafter abbreviated S.E.) is a F.S.M. of the type

\[ (\{s\}, X, Y, \Lambda, \Omega). \]

Define the function \( \Omega' \),

\[ \Omega' : X \to Y \]

by

\[ \Omega'(x) = \Omega(s, x) \]

for all \( x \in X \). In general, \( s \) and \( \Lambda \) are not important and are not saved as descriptors. A triple of the form

\[ (X, Y, \Omega') \]
denotes the S. E.  (Note that the triple notation for a S. E.  does not specify a unique F. S. M.)

The S. E., as defined above, represents a function from its input alphabet into its output alphabet.

EXAMPLE 3.

Suppose $M_1 = (S_1, X_1, Y_1, A_1, \Omega_1)$ and $M_2 = (S_2, X_2, Y_2, A_2, \Omega_2)$ are F. S. M.'s and that $R_1$ denotes the range of $\Omega_1$. Then the F. S. M. $M_1$ is said to be cascaded with the F. S. M. $M_2$ if

$$R_1 \subseteq X_2,$$

and the output from $M_1$ at every time period is taken as the input to $M_2$. As illustrated in Figure 3, the F. S. M. $(S, X, Y, A, \Omega)$ can be represented as the cascade of the O. F. M. $(S, X, A)$ and the S. E. $(S \otimes X, Y, \Omega)$.

4. THE COMPOSITION OF FINITE STATE MACHINES

4.1 Some General Remarks on Functions

DEFINITION 8.

(1) If $f : A \rightarrow B$ and $g : B \rightarrow C$, then the function

$$gf : A \rightarrow C$$

is defined by
\[ \text{gf}(a) = g(f(a)) \]

for all \( a \in A \).

(2) If \( f : A \to B \) and \( g : C \to D \), then the function

\[ [f, g] : A \otimes C \to B \otimes D \]

is defined by

\[ [f, g] (a, c) = (f(a), g(c)) \]

for all \((a, c) \in A \otimes C \).

Suppose \( f : a \to B, \ g : B \to C, \) and \( h : C \to D \). Let \( g' = gf \) and \( h' = hg \). Then

\[ h(g'(a)) = h(g(f(a))) = h'f(a) \]

for all \( a \in A \). This shows that the function composition defined in Definition 8(1) is associative, and hence that symbols of grouping may be omitted.

**LEMMA 2.**

Let \( f : A \to B, \ g : B \to C, \) and \( h : C \to D \). Then:

(1) \( f \) and \( g \) onto imply \( gf \) is onto.
(2) \( f \) and \( g \) one-one imply \( gf \) is one-one.
(3) \( f \) and \( g \) invertible imply \( gf \) is invertible.
(4) \( f \) and \( h \) onto imply \([f, h]\) is onto.
(5) \( f \) and \( h \) one-one imply \([f, h]\) is one-one.
(6) \( f \) and \( h \) invertible imply \([f, h]\) is invertible.

**Proof:**

(1) Assume \( f \) and \( g \) are onto, and \( c \in C \). Then

\[ \text{Eb} \ (b \in B \text{ and } g(b) = c), \]

and

\[ \text{Ea} \ (a \in A \text{ and } f(a) = b), \]

which implies that

\[ \text{gf}(a) = c. \]
(2) Assume \( f \) and \( g \) are one-one, and that
\[
gf(a_1) = gf(a_2)
\]
for some \( a_1 \) and \( a_2 \in A \). Then
\[
f(a_1) = f(a_2),
\]
and hence
\[
a_1 = a_2,
\]
since \( g \) and \( f \) are both one-one.

(3) The proof follows from (1) and (2) above.

(4) Assume \( f \) and \( h \) are onto, and \((b, d) \in B \times D\).
\[
E(a, c) ((a, c) \in A \times C, f(a) = b, \text{ and } h(c) = d)
\]
which implies that
\[
[f, h] (a, c) = (b, d).
\]

(5) Assume \( f \) and \( h \) are one-one, and that
\[
[f, h] (a_1, c_1) = [f, h] (a_2, c_2)
\]
for some \((a_1, c_1) \) and \((a_2, c_2) \in A \times C\).
Now
\[
(f(a_1), h(c_1)) = (f(a_2), h(c_2)),
\]
and hence
\[
a_1 = a_2 \text{ and } c_1 = c_2.
\]
This shows that
\[
(a_1, c_1) = (a_2, c_2).
\]

(6) The proof follows from (4) and (5) above.
LEMMA 3.

If $f : A \rightarrow B$, $g : B \rightarrow A$, and $A$ and $B$ are finite, then:

(1) $f$ onto implies

$\#(B) \leq \#(A)$

(2) $f$ one-one implies

$\#(A) \leq \#(B)$

(3) $f$ invertible implies

$\#(A) = \#(B)$.

(4) $\#(A) = \#(B)$ and $f$ onto imply $f$ is invertible.

(5) $\#(A) = \#(B)$ and $f$ one-one imply $f$ is invertible.

(6) $f$ and $g$ both onto implies $f$ and $g$ are both invertible.

(7) $f$ and $g$ both one-one implies $f$ and $g$ are both invertible.

Proof:

(1) Suppose $f$ is onto. If $\#(B) > \#(A)$, then

$\exists b \in B \text{ and } b \not\in f(a) \text{ for any } a \in A$,

and hence $f$ is not onto, a contradiction. Hence

$\#(B) \leq \#(A)$.

(2) Suppose $f$ is one-one. If $\#(A) > \#(B)$, then

$\exists (a_1, a_2) \in A \times A \text{ and } a_1 \neq a_2 \text{ and } f(a_1) = f(a_2)$,

and hence $f$ is not one-one, a contradiction. Hence

$\#(A) \leq \#(B)$.

(3) Assume $f$ is invertible. Then by (1) and (2) above,

$\#(A) = \#(B)$. 

(4) Assume \( \#(A) = \#(B) \) and \( f \) is onto. Suppose

\[ E(a_1, a_2) (a_1 \text{ and } a_2 \in A, \ a_1 \neq a_2, \ \text{and } f(a_1) = f(a_2)). \]

Then in order for \( f \) to be onto, it is necessary that

\[ \#(A) - 2 \geq \#(B) - 1 \]

or that

\[ \#(A) \geq \#(B) + 1, \]

a contradiction. This shows \( f \) must be one-one, and hence invertible.

(5) Assume \( \#(A) = \#(B) \) and \( f \) is one-one.

Suppose

\[ Eb (b \in B \text{ and } b \neq f(a) \text{ for any } a \in A) \]

Then in order for \( f \) to be one-one, it is necessary that

\[ \#(A) \leq \#(B) - 1, \]

a contradiction. This shows \( f \) must be onto, and hence invertible.

(6) If \( f \) and \( g \) are both onto, then

\[ \#(A) = \#(B), \]

and (6) reduces to (4) above.

(7) If \( f \) and \( g \) are both one-one, then

\[ \#(A) = \#(B) \]

and (7) reduces to (5) above.

4.2 Machine Homomorphism, Isomorphism, and Inclusion

DEFINITION 9.

Let \( M_1 = (S_1, X_1, Y_1, \Lambda_1, \Omega_1) \) and \( M_2 = (S_2, X_2, Y_2, \Lambda_2, \Omega_2) \) be F.S.M.'s. Suppose \( f : S_1 \rightarrow S_2, \ g : X_1 \rightarrow X_2, \) and \( h : Y_1 \rightarrow Y_2 \) are functions satisfying
(i) \( f \Lambda_1(s, x) = \Lambda_2(f(s), g(x)) \)

and

(ii) \( h \Omega_1(s, x) = \Omega_2(f(s), g(x)) \)

for all \( s \in S_1 \) and \( x \in X_1 \).

(1) If \( f, g, \) and \( h \) are onto, then \( M_2 \) is the homomorphic image of \( M_1 \), written as

\[
M_1 \geq M_2.
\]

(2) If \( f, g, \) and \( h \) are one-one, then \( M_1 \) is a submachine of \( M_2 \), written as

\[
M_1 \subseteq M_2.
\]

(3) If \( f, g, \) and \( h \) are invertible, then \( M_1 \) is isomorphic to \( M_2 \), written as

\[
M_1 \cong M_2.
\]

Ideas similar to at least one of the concepts in Definition 9 occur in Gill, Yoeli, Hartmanis, Ginsburg, and Rhodes. In particular, Ginsburg defines notions similar to (1), (2) and (3) above except that he does not allow recoding of the input and output alphabets through the functions \( g \) and \( h \) of Definition 9. In remarks about a partial ordering of machines according to the work they do, Rhodes suggests input and output alphabet recoding. The idea is also implicit in Hartmanis.

Suppose \( M_1 \) and \( M_2 \) are two F. S. M.'s. If \( M_1 \cong M_2 \), then \( M_1 \) and \( M_2 \) are the same F. S. M., in the sense that given a suitable recoding of input and output alphabets, \( M_1 \) and \( M_2 \) perform the same sets of functions from input strings into output strings.

If \( M_1 \geq M_2 \) or \( M_2 \subseteq M_1 \), then \( M_1 \) is at least as powerful a F. S. M. as \( M_2 \), in the sense that given a suitable recoding of \( M_2 \)'s input alphabet and \( M_1 \)'s output alphabet, \( M_1 \) can perform the same set of functions from \( M_2 \)'s input strings into \( M_2 \)'s output strings as \( M_2 \).

EXAMPLE 4.

Consider the F. S. M.'s \( M_1 \) and \( M_2 \) depicted in Figure 4. \( M_1 \) is a mod 4 adder and \( M_2 \) is a mod 2 adder. Let \( f \), \( g \), and \( h \) be functions defined as follows:

\[
g = h = \begin{pmatrix} 0 & 1 & 2 & 3 \\ 0 & 1 & 0 & 1 \end{pmatrix}
\]
and

\[ f(s_i) = t_{g(i)}, \quad 1 \leq i \leq 4 \]

It is not difficult to see that \( f, g, \) and \( h \) satisfy the conditions of Definition 9(1), and hence that

\[ M_1 \geq M_2. \]

Define two S. E. 's by

\[ E_1 = (L, L, I_L) \]

and

\[ E_2 = (Z_4, L, g). \]

If \( M_3 \) is defined to be the cascade of \( E_1 \) with \( M_1 \) with \( E_2 \), then \( M_3 \) defines the same function from input strings of \( M_2 \) to output strings of \( M_2 \) as \( M_2 \) if, when \( M_2 \) is started in state \( t_i \), \( M_1 \) is started in state \( s_i (1 \leq i \leq 2) \).

EXAMPLE 5.

If \( M_1 \) and \( M_2 \) are F. S. M's, it is not necessarily the case that
implies

\[ M_1 \geq M_2, \]

\[ M_2 \subseteq M_1, \]

or vice versa. To understand this, consider the S.F.M.'s of Figure 5. The S.F.M.'s \( F_1, F_2, \) and \( F_3 \) depicted have the following relations with each other.

\[ F_2 \geq F_1 \text{ but } F_1 \not\geq F_2, \]

\[ F_2 \subseteq F_3 \text{ but } F_3 \not\subseteq F_2, \]

and

\[ F_3 \geq F_1 \text{ and } F_1 \subseteq F_3. \]

Figure 5. \( F_1, F_2 \) and \( F_3 \) are S.F.M.'s. \( F_2 \geq F_1 \), but \( F_1 \not\geq F_2. \)

LEMMA 4.

Suppose \( M_i = (S_i, X_i, Y_i, \Lambda_i, \Omega_i) \) is a F.S.M. for \( 1 \leq i \leq 4. \)

(1) Relations \( \geq, \subseteq \) and \( \sim \) are all reflexive.
(2) Relations $\geq$, $\subseteq$, and $\simeq$ are all transitive.

(3) Relation $\approx$ is symmetric.

(4) If $M_1 \simeq M_2$, $M_3 \simeq M_4$, and $M_1 \geq M_3$, then $M_2 \geq M_4$.

(5) If $M_1 \simeq M_2$, $M_3 \simeq M_4$, and $M_1 \subseteq M_3$, then $M_2 \subseteq M_4$.

(6) If $M_1 \geq M_2$ and $M_2 \geq M_1$, then $M_1 \simeq M_2$.

(7) If $M_1 \subseteq M_2$ and $M_2 \subseteq M_1$, then $M_1 \simeq M_2$.

(8) If $\#(S_1) = \#(S_2)$, $\#(X_1) = \#(X_2)$, and $M_1 \geq M_2$, then $M_1 \simeq M_2$.

(9) If $\#(S_1) = \#(S_2)$, $\#(X_1) = \#(X_2)$, and $M_1 \subseteq M_2$, then $M_1 \simeq M_2$.

(10) If $M_1 \geq M_2$ and $M_1 \subseteq M_2$, then $M_1 \simeq M_2$.

Proof:

First the following fact will be proved. If $f : S_1 \rightarrow S_2$, $g : X_1 \rightarrow X_2$, $h : Y_1 \rightarrow Y_2$, $f' : S_2 \rightarrow S_3$, $g' : X_2 \rightarrow X_3$, and $h' : Y_2 \rightarrow Y_3$ satisfy Definition 9(i) and 9(ii), then $f'f$, $g'g$, and $h'h$ also satisfy Definition 9(i) and 9(ii). This is true because

$$f'f \Lambda_1 (s, x) = f' \Lambda_2 (f(s), g(x)) = \Lambda_3 (f'(f(s), g'g(x)))$$

and

$$h'h \Omega_1 (s, x) = h' \Omega_2 (f(s), g(x)) = \Omega_3 (f'(f(s), g'g(x))).$$

Then the proofs for Lemmas 4(1) through 4(10) are as follows:

(1) The functions $I_{S_1}$, $I_{X_1}$, and $I_{Y_1}$ fulfill the conditions of Definitions 9(1), 9(2) and 9(3). Hence reflexivity follows from

$$M_1 \geq M_1, \ M_1 \subseteq M_1, \ and \ M_1 \simeq M_1.$$  

(2) Transitivity follows from the first paragraph of this proof and Lemmas 2(1), 2(2) and 2(3).

(3) Suppose $M_1 \simeq M_2$ and $f : S_1 \rightarrow S_2$, $g : X_1 \rightarrow X_2$, and $h : Y_1 \rightarrow Y_2$ are functions satisfying Definition 9(3). Consider the functions $f^{-1}$, $g^{-1}$, and $h^{-1}$. By Definition 9(i), for all $s \in S_2$ and $x \in X_2$,

$$\Lambda_2 (s, x) = f\Lambda_1 (f^{-1}(s), g^{-1}(x)).$$
Taking $f^{-1}$ of both sides of this equation gives
\[
f^{-1} \Lambda_2(s, x) = \Lambda_1(f^{-1}(s), g^{-1}(x)).
\]
In similar fashion
\[
h^{-1} \Omega_2(s, x) = \Omega_1(f^{-1}(s), g^{-1}(x))
\]
for all $s \in S_2$ and $x \in X_2$. This shows that
\[
M_2 \cong M_1
\]
and hence that $\cong$ is symmetric.

(4) Suppose $M_1 \cong M_2$, $M_3 \cong M_4$, and $M_1 \geq M_3$. Suppose also that the invertible functions $f_1 : S_1 \rightarrow S_2$, $g_1 : X_1 \rightarrow X_2$, $h_1 : Y_1 \rightarrow Y_2$, $f_3 : S_3 \rightarrow S_4$, $g_3 : X_3 \rightarrow X_4$, and $h_3 : Y_3 \rightarrow Y_4$ and the onto functions $f : S_1 \rightarrow S_3$, $g : X_1 \rightarrow X_3$, and $h : Y_1 \rightarrow Y_3$ all satisfy Definition 9(i) and 9(ii). Lemma 2(1) and the first paragraph of this proof guarantee that the functions
\[
f_3 f_1^{-1} : S_2 \rightarrow S_4,
\]
\[
g_3 g_1^{-1} : X_2 \rightarrow X_4
\]
and
\[
h_3 h_1^{-1} : Y_2 \rightarrow Y_4
\]
are onto and satisfy Definition 9(i) and 9(ii). It follows that
\[
M_2 \geq M_4.
\]

(5) The proof is similar to (4).

(6) The proof is a direct consequence of Lemma 3(6).

(7) The proof is a direct consequence of Lemma 3(7).

(8) The proof is a direct consequence of Lemma 3(4).

(9) The proof is a direct consequence of Lemma 3(5).

(10) Assume $M_1 \geq M_2$ and $M_1 \cong M_2$. By definition 9(1) and Lemma 3(1), it follows that
\[
\#(S_1) \geq \#(S_2), \quad \#(X_1) \geq \#(S_2), \quad \text{and} \quad \#(Y_1) \geq \#(Y_2).
\]
But by Definition 9 (2) and Lemma 3(2), it follows that

\[ #(S_1) \leq #(S_2), \quad #(X_1) \leq #(X_2), \quad \text{and} \quad #(Y_1) \leq #(Y_2). \]

Hence

\[ #(S_1) = #(S_2), \quad #(X_1) = #(X_2), \quad \text{and} \quad #(Y_1) = #(Y_2). \]

Thus (10) reduces to (8).

DEFINITION 10.

(1) \( \mathcal{M} \) will be used to denote the set of all finite state machines.

(2) Define \([M_1]\) by

\[ [M_1] = \{ M \mid M \in \mathcal{M} \text{ and } M \succeq M_1 \} \]

(3) Define \( \mathcal{M}/\succeq \) by

\[ \mathcal{M}/\succeq = \{ [M_1] \mid M_1 \in \mathcal{M} \} \]

(4) Extend \( \succeq \) and \( \subseteq \) to \( \mathcal{M}/\succeq \) by defining

\[ [M_1] \succeq [M_2] \text{ if and only if } M_1 \succeq M_2 \]

and

\[ [M_1] \subseteq [M_2] \text{ if and only if } M_1 \subseteq M_2. \]

COROLLARY TO LEMMA 4.

(1) \( \succeq \) is an equivalence relation on \( \mathcal{M} \).

(2) \( \succeq \) and \( \subseteq \) are order relations on \( \mathcal{M}/\succeq \).

Proof:

(1) The proof is a consequence of Lemmas 4(1), 4(2), and 4(3).

(2) The proof is a consequence of Lemmas 4(1), 4(2), 4(4), 4(5), 4(6) and 4(7).

4.3 The Product Machine

DEFINITION 11.

If \( M_1 = (S_1, X_1, Y_1, \Lambda_1, \Omega_1) \) and \( M_2 = (S_2, X_2, Y_2, \Lambda_2, \Omega_2) \) are two F.S.M.'s, then the product of \( M_1 \) and \( M_2 \), denoted by
$M_1 \otimes M_2$

is defined to be the F. S. M.

$$(S_1 \otimes S_2, X_1 \otimes X_2, Y_1 \otimes Y_2, \Lambda, \Omega)$$

where $\Lambda : (S_1 \otimes S_2) \otimes (X_1 \otimes X_2) \rightarrow S_1 \otimes S_2$ and $\Omega : (S_1 \otimes S_2) \otimes (X_1 \otimes X_2) \rightarrow Y_1 \otimes Y_2$

are defined by

$$\Lambda((s_1, s_2), (x_1, x_2)) = (\Lambda_1(s_1, x_1), \Lambda_2(s_2, x_2))$$

and

$$\Omega((s_1, s_2), (x_1, x_2)) = (\Omega_1(s_1, x_1), \Omega_2(s_2, x_2))$$

for all $s_1 \in S_1, s_2 \in S_2, x_1 \in X_1,$ and $x_2 \in X_2$.

Note that in Definition 11, $M_1 \otimes M_2$ is indeed a F. S. M. since Lemma 1(2) guarantees that $S_1 \otimes S_2, X_1 \otimes X_2$ and $Y_1 \otimes Y_2$ are all alphabets.

If $M_1$ and $M_2$ are F. S. M.'s, $M_1 \otimes M_2$ is essentially the F. S. M. consisting of $M_1$ and $M_2$ operating simultaneously. Specification of a starting state for $M_1 \otimes M_2$ consists in the specification of a starting state for $M_1$ and a starting state for $M_2$. An input letter to $M_1 \otimes M_2$ consists of an input letter for $M_1$ and an input letter for $M_2$, while an output letter from $M_1 \otimes M_2$ consists of an output letter from $M_1$ and an output letter from $M_2$.

EXAMPLE 6.

The F. S. M.'s $M_1$, $M_2$, and $M_3$ of Figure 6 are respectively a mod 3 adder, a mod 2 adder, and a mod 6 adder. Let

$$g = h = \begin{pmatrix}
(0, 0) & (1, 1) & (2, 0) & (0, 1) & (1, 0) & (2, 1) \\
0 & 1 & 2 & 3 & 4 & 5
\end{pmatrix}$$

and

$$f(s_i, t_j) = u_{g(i, j)}$$

for $1 \leq i \leq 3$ and $1 \leq j \leq 2$. The functions $f$, $g$, and $h$ are from the state, input, and output alphabets, respectively, of $M_1 \otimes M_2$ into the state, input, and output alphabets, respectively, of $M_3$. These functions satisfy Definition 9(3), and hence

$$M_1 \otimes M_2 \simeq M_3$$
Figure 6. $M_1$ is a Mod 3 Adder, $M_2$ is a Mod 2 Adder, and $M_3$ is a Mod 6 Adder. $M_3 \equiv M_1 \otimes M_2$

EXAMPLE 7.

Suppose $M_1 = (S_1, L, L, \Lambda_1, \Omega_1)$ and $M_2 = (S_2, L, L, \Lambda_2, \Omega_2)$ are two F. S. M. 's. Suppose also that $E = (L \otimes L, L, \Omega)$ is a S. E. where

$$\Omega(x_1, x_2) = \begin{cases} 1 & \text{if } x_1 = x_2 \\ 0 & \text{otherwise} \end{cases}$$

If $M_3$ is the cascade of $M_1 \otimes M_2$ and $E$, and all inputs to $M_3$ are of the form $(x, x)$,

then the output string from $M_3$ is a string of 1's if and only if $M_1$ and $M_2$, in their respective starting states, respond in identical manners to identical input strings.
EXAMPLE 8.

Suppose $M_1 = (S_1, X, W, \Lambda_1, \Omega_1)$ and $M_2 = (S_2, W, Y, \Lambda_2, \Omega_2)$ are F. S. M.'s. Define the machine

$$M_3 = (S_1 \otimes S_2, X, Y, \Lambda, \Omega)$$

where

$$\Lambda((s_1, s_2), x) = (\Lambda_1(s_1, x), \Lambda_2(s_2, \Omega_1(s_1, x)))$$

and

$$\Omega((s_1, s_2), x) = \Omega_2(s_2, \Omega_1(s_1, x))$$

for all $s_1 \in S_1$, $s_2 \in S_2$, and $x \in X$. The F. S. M. $M_3$ is the cascade of $M_1$ with $M_2$. Though the alphabet $W$ was suppressed in the description of $M_3$, it is evident that when it is considered, $M_3$ is just $M_1 \otimes M_2$ where the input letter to $M_2$ must be the output letter from $M_1$.

DEFINITION 12.

Define the F. S. M. $i$ by

$$i = (\{1\}, \{1\}, \{1\}, \Lambda, \Omega)$$

where

$$\Lambda(1, 1) = \Omega(1, 1) = 1.$$

LEMMA 5.

Suppose $M_1 = (S_1, X_1, Y_1, \Lambda_1, \Omega_1)$ is a F. S. M. for $1 \leq i \leq 4$.

1. $M_1 \geq M_2$ and $M_2 \geq M_4$ imply $M_1 \otimes M_3 \geq M_2 \otimes M_4$
2. $M_1 \subseteq M_2$ and $M_2 \subseteq M_4$ imply $M_1 \otimes M_3 \subseteq M_2 \otimes M_4$
3. $M_1 \cong M_2$ and $M_3 \cong M_4$ imply $M_1 \otimes M_3 \cong M_2 \otimes M_4$
4. $M_1 \otimes (M_2 \otimes M_3) \cong (M_1 \otimes M_2) \otimes M_3$
5. $M_1 \otimes M_2 \cong M_2 \otimes M_1$
6. $\emptyset \otimes M_1 \cong M_1$
7. It is not true that $M_1 \otimes M_2 \cong M_1 \otimes M_3$ implies $M_2 \cong M_3$.

*This was suggested by C. L. Liu.
Proof

First the following fact will be proved. If \( f : S_1 \to S_2, \ g : X_1 \to X_2, \)
\( h : Y_1 \to Y_2, \ f' : S_3 \to S_4, \ g' : X_3 \to X_4, \) and \( h' : Y_3 \to Y_4 \) satisfy Definition 9(i) and
9(ii), then so do \([f,f'], [g,g'], [h,h']\) with respect to the functions \( \Lambda, \Omega, \Lambda', \)
and\( \Omega' \) defined by

\[
M_1 \otimes M_3 = (S_1 \otimes S_3, X_1 \otimes X_3, Y_1 \otimes Y_3, \Lambda, \Omega)
\]

and

\[
M_2 \otimes M_4 = (S_2 \otimes S_4, X_2 \otimes X_4, Y_2 \otimes Y_4, \Lambda', \Omega').
\]

This is true because

\[
[f,f'] \Lambda ((s_1, s_3), (x_1, x_3)) = [f,f'] (\Lambda_1(s_1, x_1), \Lambda_3(s_3, x_3)) = ((\Lambda_1(s_1, x_1), f'(s_3)), \Lambda_3(s_3, x_3)) = (\Lambda_2((f(s_1), g(x_1)), \Lambda_4(f'(s_3), g'(x_3))) = \Lambda'((f(s_1), f'(s_3)), (g(x_1), g'(x_3))) = \Lambda'([f,f'](s_1, s_3), [g,g'](x_1, x_3))
\]

for all \( s_1 \in S_1, s_3 \in S_3, x_1 \in X_1, \) and \( x_3 \in X_3. \)

A similar proof follows for \( \Omega \) and \( \Omega'. \)

Then the proofs for Lemmas 5(1) through 5(7) are as follows:

1. The proof follows from Lemma 2(4) and the first part of the proof above.
2. The proof follows from Lemma 2(5) and the first part of the proof above.
3. The proof follows from Lemma 2(6) and the first part of the proof above.
4. Let \( f : S_1 \otimes (S_2 \otimes S_3) \to (S_1 \otimes S_2) \otimes S_3, \ g : X_1 \otimes (X_2 \otimes X_3) \to (X_1 \otimes X_2) \otimes X_3, \)
and \( h : Y_1 \otimes (Y_2 \otimes Y_3) \to (Y_1 \otimes Y_2) \otimes Y_3 \) be defined by

\[
f(s_1, (s_2, s_3)) = ((s_1, s_2), s_3),
\]
\[
g(x_1, (x_2, x_3)) = ((x_1, x_2), x_3),
\]

and

\[
h(y_1, (y_2, y_3)) = ((y_1, y_2), y_3).
\]
Clearly $f$, $g$, and $h$ are invertible, for all pairs in their domains, and it is not difficult to see that they satisfy Definition 9(i) and 9(ii).

(5) Let $f : S_1 \otimes S_2 \to S_1 \otimes S_1$, $g : X_1 \otimes X_2 \to X_2 \otimes X_1$ and $h : Y_1 \otimes Y_2 \to Y_2 \otimes Y_1$ be defined by

$$f(s_1, s_2) = (s_2, s_1)$$
$$g(x_1, x_2) = (x_2, x_1)$$

and

$$h(y_1, y_2) = (y_2, y_1)$$

for all $s_1 \in S_1$, $s_2 \in S_2$, $x_1 \in X_1$, $x_2 \in X_2$, $y_1 \in Y_1$ and $y_2 \in Y_2$. The functions $f$, $g$, and $h$ satisfy Definition 9(3) for isomorphism.

(6) Let $f : \{1\} \otimes S_1 \to S_1$, $g : \{1\} \otimes X_1 \to X_1$, and $h : \{1\} \otimes Y_1 \to Y_1$ be defined by

$$f(1, s_1) = s_1$$
$$g(1, x_1) = x_1$$

and

$$h(1, y_1) = y_1$$

for all $s_1 \in S_1$, $x_1 \in X_1$, and $y_1 \in Y_1$. Clearly $f$, $g$, and $h$ satisfy the conditions of Definition 9(3) for isomorphism.

(7) Figure 7 shows S. F. M.'s $M_1$, $M_2$, and $M_3$ having the properties

$$M_2 \otimes M_2 \cong M_2 \otimes M_1$$

but

$$M_1 \not\cong M_2.$$
Figure 7. $M_1$, $M_2$ and $M_3$ are S.F.M. 's. $M_2 \oplus M_1 \cong M_3 \cong M_2 \oplus M_2$ but $M_1 \not\cong M_2$.

In general, the reason for investigating F.S.M. multiplication is that if a given F.S.M. is isomorphic to the product of two smaller F.S.M. 's, it may well be desirable to use the two smaller F.S.M. 's to synthesize the original F.S.M. The problem is to tell when a given F.S.M. is decomposable.

If $(\mathbb{M}/\mathfrak{E}, \otimes)$ were a submonoid of a group, then for any two machines, $M_1$ and $M_2$, it could be determined whether

$$EM_3(M_3 \otimes M_2 \cong M_1)$$

by taking the product of $M_1$ and $M_2^{-1}$. If $M_1 \otimes M_2^{-1}$ were a F.S.M., then

$$(M_1 \otimes M_2^{-1}) \otimes M_2 \cong M_1,$$

and $M_1 \otimes M_2^{-1}$ would be the desired F.S.M. $M_3$. If, on the other hand, $M_1 \otimes M_2^{-1}$ were not a F.S.M., then it would not be the case that

$$EM_3(M_3 \otimes M_2 = M_1).$$

Since, however, $(\mathbb{M}/\mathfrak{E}, \otimes)$ cannot be a submonoid of any group, a checking scheme of the above nature cannot be found.
Some Theorems on the Decomposability of a Finite State Machine

THEOREM 1.

Suppose $M_i = (S_i, X_i, Y_i, A_i, \Omega_i)$ is a F.S.M. for $1 \leq i \leq 3$. Then

$$M_3 \cong M_1 \otimes M_2$$

if and only if

1. $M_3 \subseteq M_1 \otimes M_2$
2. $\#(S_1) \cdot \#(S_2) = \#(S_3)$, $\#(X_1) \cdot \#(X_2) = \#(X_3)$, and $\#(Y_1) \cdot \#(Y_2) = \#(Y_3)$.

Proof:

The following is used in the proof.

(i) $\#(A \otimes B) = \#(A) \cdot \#(B)$

for all finite sets $A$ and $B$. The proof will be divided into two parts.

1. Assume $M_3 \cong M_1 \otimes M_2$. Then: Part (1) of Theorem 1 is obvious; and part (2) of Theorem 1 follows from Definition 9(4), Lemma 3(3) and (i).

2. Assume (1) and (2) hold. By (i) and Lemma 4(9)

$$M_3 \cong M_1 \otimes M_2$$

THEOREM 2.

Suppose $d: A \rightarrow B$. Define $d^{-1}: 2^B \rightarrow 2^A$ by $d^{-1}(B') = \{a \mid a \in A, b' = f(a), \text{ and } b' \in B\}$.

Suppose $M_i = (S_i, X_i, Y_i, A_i, \Omega_i)$ is a F.S.M. for $1 \leq i \leq 3$. Then

$$M_3 \cong M_1 \otimes M_2$$

implies

1. $M_3 \supseteq M_1$ and $M_3 \supseteq M_2$

2. If $f: S_3 \rightarrow S_1 \otimes S_2$, $g: X_3 \rightarrow X_1 \otimes X_2$, and $h: Y_3 \rightarrow Y_1 \otimes Y_2$, satisfy Definition 9(3) for isomorphism, define $f_1: S_3 \rightarrow S_1$, $f_2: S_3 \rightarrow S_2$, $g_1: X_3 \rightarrow X_1$, $g_2: X_3 \rightarrow X_2$, $h_1: Y_3 \rightarrow Y_1$, and $h_2: Y_3 \rightarrow Y_2$ by

$$f(a) = (f_1(a), f_2(a)),$$
$$g(x) = (g_1(x), g_2(x)),$$
$$h(y) = (h_1(y), h_2(y)).$$
and
\[ h(y) = (h_1(y), h_2(y)) \]
for all \( s \in S_3, x \in X_3, \) and \( y \in Y_3 \). Then
\[ \#(f_1^{-1}(s)) = \#(S_1) \text{ and } \#(f_2^{-1}(s)) = \#(S_2) \]
\[ \#(g_1^{-1}(x)) = \#(X_1) \text{ and } \#(g_2^{-1}(x)) = \#(X_2) \]
and
\[ \#(h_1^{-1}(y)) = \#(Y_1) \text{ and } \#(h_2^{-1}(y)) = \#(Y_2) \]
for all \( s_1 \in S_1, s_2 \in S_2, x_1 \in X_1, x_2 \in X_2, y_1 \in Y_1, \) and \( y_2 \in Y_2 \).

Proof:
Assume \( M_3 \cong M_1 \otimes M_2 \) and let \( f, g, h, f_1, f_2, g_1, g_2, h_1, \) and \( h_2 \)
be the functions defined in part (2) of Theorem 2.

(1) Since \( f, g, \) and \( h \) are onto, it follows that \( f_1, f_2, g_1, g_2, h_1, \) and \( h_2 \)
are onto. It is also true that for all \( s_3 \in S_3 \) and \( x_3 \in X_3 \)
\[ (f_1 \Lambda_3 (s_3, x_3), f_2 \Lambda_3 (s_3, x_3)) \]
\[ = f \Lambda_3 (s_3, x_3) \]
\[ = \Lambda_3 (f(s_3), g(x_3)) \]
\[ = \Lambda_3 ((f_1(s_3), f_2(s_3)), (g_1(x_3), g_2(x_3))) \]
\[ = (\Lambda_1 (f_1(s_3), g_1(x_3)), \Lambda_2 (f_2(s_3), g_2(x_3))) \],
and similarly that for all \( s_3 \in S_3 \) and \( x_3 \in X_3 \)
\[ (h_1 \Omega_3 (s_3, x_3), h_2 \Omega_3 (s_3, x_3)) \]
\[ = (\Omega_1 (f_1(s_3), g_1(x_3)), \Omega_2 (f_2(s_3), g_2(x_3))) \].

It follows that
\[ M_3 \cong M_1 \text{ and } M_3 \cong M_2. \]
(2) Suppose \( s_1 \in S_1 \). Then

\[
\hat{f}_1^{-1}(s_1) = \{ s \mid s \in S_3 \text{ and } f_1(s) = s_1 \}
\]

\[
= \{ s \mid s \in S_3 \text{ and } f(s) \in \{ s_1 \} \otimes S_2 \}. 
\]

Since \( f \) is invertible, it is not difficult to see that

\[
\#(\hat{f}_1^{-1}(s_1)) = \#(\{ s_1 \} \otimes S_2) = \#(S_2).
\]

**DEFINITION 13.**

Let \( M = (S, X, Y, A, \Omega) \) by a F.S.M. Suppose \( \sim, =, \text{ and } \simeq \) are equivalence relations on \( S, X, \) and \( Y, \) respectively; and that \( S^+, X^+, \) and \( Y^+ \) denote, respectively, the set of equivalence classes of \( S \) under \( \sim, \) the set of equivalence classes of \( X \) under \( =, \) and the set of equivalence classes of \( Y \) under \( \simeq. \)

If

\[
s \sim s' \text{ and } x \simeq x'
\]

implies

\[
\Lambda(s, x) \sim \Lambda(s', x')
\]

and

\[
\Omega(s, x) \simeq \Omega(s', x')
\]

for all \( s \) and \( s' \in S \) and \( x \) and \( x' \in X, \) then \( M \) is said to be partitioned by a partition with substitution property. In this case two functions

\[
\Lambda^+: S^+ \otimes X^+ \rightarrow S^+
\]

and

\[
\Omega^+: S^+ \otimes X^+ \rightarrow Y^+
\]

may be defined by

\[
\Lambda^+(S_i, X_j) = \{ s_m \mid s_m \sim \Lambda(s_i, x_j) \text{ and } (s_i, x_j) \in S_i \otimes X_j \}
\]
and

\[ \Omega^+(s_i, x_j) = \{ y_n | y_n \in \Omega(s_i, x_j) \text{ and } (s_i, x_j) \in S_i \otimes X_j \} \]

for all \( s_i \in S^+ \) and \( x_j \in X^+ \).

Note in Definition 13 that if \( s_i \in S^+ \) and \( x_j \in X^+ \), then

\[ \Lambda^+(s_i, x_j) \in S^+ \] and \( \Omega^+(s_i, x_j) \in Y^+ \]

because of the substitution property. Note also that

\[ M^+ = (S^+, X^+, Y^+, \Lambda^+, \Omega^+) \]

the F.S.M. induced by the partition with substitution property on \( M \) is indeed a F.S.M., since \( S^+ \), \( X^+ \), and \( Y^+ \) are alphabets by Lemmas 1(1) and 1(3).

**LEMMA 6.**

Let \( M = (S, X, Y, \Lambda, \Omega) \) be a F.S.M.

1. If \( M^+ = (S^+, X^+, Y^+, \Lambda^+, \Omega^+) \) is a F.S.M. induced by a partition with substitution property on \( M \), then

\[ M \cong M^+ \]

2. If \( M' = (S', X', Y', \Lambda', \Omega') \) is a F.S.M. and

\[ M \geq M' \]

then there exists a F.S.M. \( M^+ = (S^+, X^+, Y^+, \Lambda^+, \Omega^+) \) induced by a partition with substitution property on \( M \) s.t.

\[ M' \cong M^+. \]

**Proof:**

1. Define \( f : S \rightarrow S^+ \), \( g : X \rightarrow X^+ \), and \( h : Y \rightarrow Y^+ \) by

\[ f(s_i) = S_i \text{ where } S_i \in S^+ \text{ and } s_i \in S_i, \]

\[ g(x_j) = X_j \text{ where } X_j \in X^+ \text{ and } x_j \in X_j, \]

and

\[ h(y_k) = Y_k \text{ where } Y_k \in Y^+ \text{ and } y_k \in Y_k \]
for all \( s_i \in S, \ x_j \in X, \) and \( y_k \in Y. \) Functions \( f, \ g, \) and \( h \) are well defined since no element is contained in two equivalence classes. Functions \( f, \ g, \) and \( h \) are onto since every equivalence class must have at least one member. Notice that

\[
f \Lambda (s, x) = S_i \text{ where } S_i \in S^+ \text{ and } \Lambda (s, x) \in S_i
\]

and

\[
h \Omega (s, x) = Y_k \text{ where } Y_k \in Y^+ \text{ and } \Omega (s, x) \in Y_k
\]

for all \( s \in S \) and \( x \in X. \) This shows that

\[M \geq M^+.\]

(2) Suppose \( M \geq M' \) and \( f : S \rightarrow S', \ g : X \rightarrow X', \) and \( h : Y \rightarrow Y' \) satisfy Definition 9(1). Define the relations \( \sim \) on \( S, \approx \) on \( X, \) and \( \cong \) on \( Y \) by

\[
s_1 \sim s_2 \text{ if and only if } f(s_1) = f(s_2)
\]

\[
x_1 \cong x_2 \text{ if and only if } g(x_1) = g(x_2)
\]

\[
y_1 \cong y_2 \text{ if and only if } h(y_1) = h(y_2)
\]

for all \( s_1, s_2 \in S, \ x_1 \text{ and } x_2 \in X, \) and \( y_1 \text{ and } y_2 \in Y. \) It is not difficult to see that all of these relations are equivalence relations. Suppose \( s_1 \text{ and } s_2 \in S, \ x_1 \text{ and } x_2 \in X, \) \( s_1 \sim s_2, \) and \( x_1 \sim x_2. \) Then

\[
f \Lambda (s_1, x_1) = \Lambda' (f(s_1), g(x_1))
\]

\[
= \Lambda' (f(s_2), g(x_2)) = f \Lambda (s_2, x_2)
\]

and

\[
h \Omega (s_1, x_1) = \Omega' (f(s_1), g(x_1))
\]

\[
= \Omega' (f(s_2), g(x_2)) = h \Omega (s_2, x_2)
\]
together imply
\[ \Lambda(s_1, x_1) \sim \Lambda(s_2, x_2) \]

and
\[ \Omega(s_1, x_1) \simeq \Omega(s_2, x_2) \]

and hence the partition on \( M \) induced by \( \sim, \approx, \) and \( \simeq \) is a partition with substitution property. Let \( M^+ = (S^+, S^+, Y^+, \Lambda^+, \Omega^+) \) be the F.S.M. induced by this partition, and let \( f' : S' \rightarrow S^+, \ g' : X' \rightarrow X^+, \) and \( h' : Y' \rightarrow Y^+ \) be defined by

\[
f'(s') = \{ s \mid s \in S \text{ and } f(s) = s' \},
\]

\[
g'(x') = \{ x \mid x \in X \text{ and } g(x) = x' \},
\]

and
\[
h'(y') = \{ y \mid y \in Y \text{ and } h(y) = y' \}
\]

for all \( s' \in S', \ x' \in X', \) and \( y' \in Y'. \) Functions \( f', \ g', \) and \( h' \) are defined for all members of \( S', \ X', \) and \( Y', \) because \( f, g, \) and \( h \) are onto. Functions \( f', \ g', \) and \( h' \) are invertible as a direct consequence of their definitions and the definitions of \( \sim, \approx, \) and \( \simeq. \) Suppose \( s' \in S', \ x' \in X', \ s \in f'(s'), \) and \( x \in g'(x'). \) Then since \( M \geq M' \)

\[
 f' \Lambda(s, x) = \Lambda'(s', x').
\]

This implies
\[ \Lambda(s, x) \subseteq f' \Lambda'(s', x') \]

which in turn implies
\[ f' \Lambda'(s', x') = \Lambda^+(f'(s'), g'(x')). \]

Similarly
\[ h' \Omega'(s', x') = \Omega^+(f'(s'), g'(x')). \]

This shows that
\[ M' \cong M^+. \]
If $M$ is a F.S.M., Lemma 6 shows that the idea of a partition on $M$ with substitution property is equivalent to the idea of a homomorphic image of $M$. Hartmanis uses the idea of partition with substitution property in the analysis of F.S.M. decomposition. Hartmanis' Theorem 11 would hold, even if all alphabets associated with a F.S.M. were allowed to be infinite. Such a strong theorem is unnecessary, and the conditions which it contains are somewhat difficult to check. Theorems 1 and 2 of this work take advantage of the finiteness of the alphabets associated with a F.S.M. Part (2) of Theorem 2 was suggested by a similar theorem in Yoeli.

4.5 The Sum Finite State Machine

**Definition 14.**

Suppose $M_1 = (S_1, X_1, Y_1, \Lambda_1, \Omega_1)$ and $M_2 = (S_2, X_2, Y_2, \Lambda_2, \Omega_2)$ are F.S.M.'s and that the following conditions are satisfied:

1. $S_1 \cup S_2, X_1 \cup X_2$, and $Y_1 \cup Y_2$ are alphabets.
2. $(S_1 \cup S_2) \otimes (X_1 \cup X_2) = (S_1 \otimes X_1) \cup (S_2 \otimes X_2)$.
3. $(S_1 \otimes X_1) \cap (S_2 \otimes X_2) = \emptyset$.

The sum of $M_1$ and $M_2$, denoted by

$$M_1 + M_2$$

is defined to be the F.S.M.

$$(S_1 \cup S_2, X_1 \cup X_2, Y_1 \cup Y_2, \Lambda, \Omega)$$

where $\Lambda : (S_1 \cup S_2) \otimes (X_1 \cup X_2) \rightarrow S_1 \cup S_2$ and $\Omega : (S_1 \cup S_2) \otimes (X_1 \cup X_2) \rightarrow Y_1 \cup Y_2$ are defined by

$$\Lambda(s, x) = \begin{cases} 
\Lambda_1(s, x) & \text{if } (s, x) \in S_1 \otimes X_1 \\
\Lambda_2(s, x) & \text{if } (s, x) \in S_2 \otimes X_2 
\end{cases}$$

and

$$\Omega(s, x) = \begin{cases} 
\Omega_1(s, x) & \text{if } (s, x) \in S_1 \otimes X_1 \\
\Omega_2(s, x) & \text{if } (s, x) \in S_2 \otimes X_2 
\end{cases}$$

Suppose $M_1 = (S_1, X_1, Y_1, \Lambda_1, \Omega_1)$ and $M_2 = (S_2, X_2, Y_2, \Lambda_2, \Omega_2)$ s.t. parts (1), (2) and (3) of Definition 14 are satisfied and $S_1, S_2, X_1, X_2$ are not empty. Part (3) implies either $S_1 \cap S_2 = \emptyset$ or $X_1 \cap X_2 = \emptyset$. Assume $S_1 \cap S_2 = \emptyset$. Intersecting $S_1 \otimes X_1$ with both sides of (2) above gives
and this in turn implies

\[ X_2 \subseteq X_1. \]

Intersecting \( S_j \otimes X_j \) with both sides of (2) above would allow one to show \( X_1 \subseteq X_2 \).

Hence

\[ X_1 = X_2. \]

Similarly when \( X_1 \cap X_2 = \emptyset \), it must be the case that \( S_1 = S_2 \).

**EXAMPLE 9.**

In Figure 8, \( M_1, M_2, M_3, M_4, \) and \( M_5 \) are O.F.M.'s and

\[ M_1 = M_2 + M_3 = M_4 + M_5. \]

Notice that the state alphabets of \( M_2 \) and \( M_3 \) are identical and that their input alphabets are disjoint. On the other hand, the input alphabets of \( M_4 \) and \( M_5 \) are identical and their state sets are disjoint.

**LEMMA 7.**

Suppose \( M_1 = (S_1, X_1, Y_1, \Lambda_1, \Omega_1) \) is a F.S.M. for \( 1 \leq i \leq 4 \). Suppose also that \( M_1 + M_j \) is defined for \( 1 \leq i < j \leq 3 \).

(1) If \( M_1 + (M_2 + M_3) \) is defined, then \( (M_1 + M_2) + M_3 \) is defined and

\[ M_1 + (M_2 + M_3) = (M_1 + M_2) + M_3. \]

(2) \( M_2 + M_1 \) is defined and \( M_1 + M_2 = M_2 + M_1 \).

(3) \( M_4 = M_1 + M_2 \) implies \( M_1 \subseteq M_4 \) and \( M_2 \subseteq M_4 \).

**Proof:**

The proof is not difficult and hence is omitted.

**THEOREM 3.**

Suppose \( M_1 = (S_1, X_1, Y_1, \Lambda_1, \Omega_1) \) is a F.S.M. for \( 1 \leq i \leq 3 \). Suppose also that \( M_2 + M_3 \) is defined. Then \( (M_1 \otimes M_2) + (M_1 \otimes M_3) \) is defined, and

\[ M_1 \otimes (M_2 + M_3) = (M_1 \otimes M_2) + (M_1 \otimes M_3). \]
Figure 8. $M_1$, $M_2$, $M_3$, $M_4$ and $M_5$ are O.F. M.'s. $M_1 = M_2 + M_3 = M_4 + M_5$

Proof:
(1) Suppose $(s_1, s_2, x_1, x_2) \in (S_1 \otimes S_2) \otimes (X_1 \otimes X_2)$. Then

$$(s_2, x_2) \notin S_3 \otimes X_3$$

and it follows that

$$(s_1, s_2, x_1, x_2) \notin (S_1 \otimes S_3) \otimes (X_1 \otimes X_3).$$

Hence

$$((S_1 \otimes S_2) \otimes (X_1 \otimes X_2)) \cap ((S_1 \otimes S_3) \otimes (X_1 \otimes X_3)) = \emptyset.$$
Now suppose \((s_1, s, x_1, x) \in (S_1 \otimes S_3) \otimes (X_1 \otimes X_3)\). Then
\[(s, x) \in (S_2 \cup S_3) \otimes (X_2 \cup X_3)\]
is implied by
\[(s, x) \in (S_2 \otimes X_2) \cup (S_3 \otimes X_3),\]
which in turn implies
\[(s_1, s, x_1, x) \in (S_1 \otimes S_2) \otimes (X_1 \otimes X_2) \cup (S_1 \otimes S_3) \otimes (X_1 \otimes X_3).\]
It follows that
\[
(S_1 \otimes S_2) \cup (S_1 \otimes S_3) \otimes (X_1 \otimes X_2) \cup (X_1 \otimes X_3)
= (S_1 \otimes S_2) \otimes (X_1 \otimes X_2) \cup (S_1 \otimes S_3) \otimes (X_1 \otimes X_3).
\]
This shows that \((M_1 \otimes M_2) + (M_1 \otimes M_3)\) is defined.

(2) Suppose \(M_1 \otimes (M_2 + M_3) = (S, X, Y, \Lambda, \Omega)\) and \((M_1 \otimes M_2) + (M_1 \otimes M_3) = (S, X, Y, \Lambda', \Omega')\) where \(S = S_1 \otimes (S_2 \cup S_3), X = X_1 \otimes (X_2 \cup X_3)\) and \(Y = Y_1 \otimes (Y_2 \cup Y_3)\). Then
\[
\Lambda((s_1, s), (x_1, x)) = \begin{cases} 
\Lambda_1(s_1, x_1), \Lambda_2(s, x) & \text{if } (s, x) \in S_2 \otimes X_2 \\
\Lambda_1(s_1, x_1), \Lambda_3(s, x) & \text{if } (s, x) \in S_3 \otimes X_3 
\end{cases}
\]
for all \(s_1 \in S_1, s \in S_2 \cup S_3, x_1 \in X_1,\) and \(x \in X_2 \cup X_3\). Similarly
\[
\Omega((s_1, s), (x_1, x)) = \Omega'((s_1, s), (x_1, x))
\]
for all \(s_1 \in S_1, s \in S_2 \cup S_3, s_1 \in X_1,\) and \(x \in X_2 \cup X_3\). If follows that
\[
M_1 \otimes (M_2 + M_3) = (M_1 \otimes M_2) + (M_1 \otimes M_3)
\]
Suppose that \(M_i\) and \(M_j\) are F.S.M.'s for \(1 \leq i \leq m\) and \(1 \leq j \leq n\), and that \(M = M_1 + M_2 + \ldots + M_m\) and \(N = N_1 + N_2 + \ldots + N_n\) are defined. Theorem 3 shows that
\[
M \otimes N = \sum_{i=1}^{m} \sum_{j=1}^{n} (M_i \otimes N_j).
\]
4.6 Output Free Machines

Suppose \( M = (S, X, Y, \Lambda, \Omega) \) is a F.S.M. and \( \Omega \) is onto. (If \( \Omega \) is not onto, consider \( M' = (S, X, Y', \Lambda, \Omega) \) where \( Y' \) is the range of \( \Omega \). Clearly \( M \) and \( M' \) are essentially the same F.S.M.) If \( N = (S, X, \Lambda) \) is an O.F.M., then it is not difficult to see that

\[ N \triangleright M. \]

In the sense of the discussion preceding Example 4, \( N \) is at least as powerful as \( M \). In view of this fact, a few results concerning the decomposition of O.F.M.'s will be presented in paragraph 4.6. These results will also serve as the motivation for studying S.F.M.'s in section 5.

**LEMMA 8.**

Suppose \( N_i = (S_i, X_i, \Lambda_i) \) is an O.F.M. for \( 1 \leq i \leq 2 \).

1. Suppose \( f : S_1 \rightarrow S_2 \) and \( g : X_1 \rightarrow X_2 \) are s.t.

\[ fA_1(s_1, x_1) = A_2(f(s_1), g(x_1)) \]

for all \( s_1 \in S_1 \) and \( x_1 \in X_1 \). Then:

   a. If \( f \) and \( g \) are onto, \( N_1 \triangleright N_2 \).
   b. If \( f \) and \( g \) are one-one, \( N_1 \cong N_2 \).
   c. If \( f \) and \( g \) are invertible, \( N_1 \cong N_2 \).

2. \( N_1 \otimes N_2 \cong (S_1 \otimes S_2, X_1 \otimes X_2, \Lambda) \) where

\[ \Lambda ((s_1, s_2), (x_1, x_2)) = \Lambda_1(s_1, x_1) \Lambda_2(s_2, x_2) \]

for all \( s_1 \in S_1, s_2 \in S_2, x_1 \in X_1, \) and \( x_2 \in X_2 \).

3. If \( N_1 + N_2 \) is defined, \( N_1 + N_2 = (S_1 \cup S_2, X_1 \cup X_2, \Gamma) \) where

\[ \Gamma(s, x) = \begin{cases} \Lambda_1(s, x) & \text{if } (s, x) \in S_1 \otimes X_1 \\ \Lambda_2(s, x) & \text{if } (s, x) \in S_2 \otimes X_2 \end{cases} \]

**Proof:**

1. Assume \( f \) and \( g \) satisfy part (1)(a) of Lemma 8.
   a. Then \( [f, g] : S_1 \otimes X_1 \rightarrow S_2 \otimes X_2 \) is onto. Furthermore

\[ [f, g]_1 \Phi X_1 (s_1, x_1) = [f, g]_2 \Phi X_1 (f(s_1), g(x_1)) \]

   = \( \{((s_1), g(x_1)) \} = \Phi_2 \Phi X_1 (f(s_1), g(x_1)). \)
This shows that \( f, \ g, \) and \([f, g]\) satisfy Definition 9 (i), and it follows that they satisfy the homomorphism conditions of Definition 9(1).

(b) Proof similar to (a) above.
(c) Proof similar to (a) above.

(2) \( N_1 \circ N_2 = (S_1 \circ S_2, X_1 \circ X_2, Y, \Lambda, I_Y) \) where

\[
Y = (S_1 \circ X_1) \circ (S_2 \circ X_2).
\]

Let \( h : (S_1 \circ X_1) \circ (S_2 \circ X_2) \rightarrow (S_1 \circ S_2) \circ (X_1 \circ X_2) \) be defined by

\[
h((s_1, x_1), (s_2, x_2)) = ((s_1, s_2), (x_1, x_2)).
\]

for all \( s_1 \in S_1, \ s_2 \in S_2, \ x_1 \in X_1, \) and \( x_2 \in X_2. \) The functions \( I_{S_1 \circ S_2}, I_{X_1 \circ X_2}, \) and \( h \) are clearly invertible and satisfy Definition 9 (ii). It is not difficult to see that they also satisfy Definition 9 (ii), and hence that

\[
N_1 \circ N_2 \equiv (S_1 \circ S_2, X_1 \circ X_2, \Lambda).
\]

(3) \( N_1 + N_2 = (S_1 \cup S_2, X_1 \cup X_2, Y, \Gamma, I_Y) \) where

\[
Y = (S_1 \circ X_1) \cup (S_2 \circ X_2).
\]

By part (2) of Definition 14

\[
Y = (S_1 \cup S_2) \circ (X_1 \cup X_2),
\]

and it follows that

\[
N_1 + N_2 = (S_1 \cup S_2, X_1 \cup X_2, \Gamma).
\]

**LEMMA 9.**

Suppose \( F_1 = (S_1, x, \Lambda_1) \) and \( F_2 = (S_2, x, \Lambda_2) \) are S.F.M.'s.

(1) Suppose \( f : S_1 \rightarrow S_2 \) and

\[
f\Lambda_1(s_1) = \Lambda_2 f(s_1)
\]

for all \( s_1 \in S_1. \) Then:

(a) If \( f \) is onto, \( F_1 \geq F_2. \)
(b) If \( f \) is one-one, \( F_1 \subseteq F_2. \)
(c) If \( f \) is invertible, \( F_1 \cong F_2 \).

(2) \( F_1 \otimes F_2 \cong (S_1 \otimes S_2, (x_1, x_2), \Lambda) \) where

\[
\Lambda(s_1, s_2) = (\Lambda_1(s_1), \Lambda_2(s_2))
\]

for all \( s_1 \in S_1 \) and \( s_2 \in S_2 \).

Proof:
(1) The O.F.M. representations of \( F_1 \) and \( F_2 \) are \( F_1 = (S_1, \{x_1\}, \Lambda_1') \) and \( F_2 = (S_2, \{x_2\}, \Lambda_2) \) where

\[
\Lambda_1'(s_1, x_1) = \Lambda_1(s_1)
\]

for all \( s_1 \in S_1 \), where \( 1 \leq 1 \leq 2 \).

(a) Let \( g : \{x_1\} \rightarrow \{x_2\} \) be defined by

\[
g(x_1) = x_2.
\]

Clearly \( g \) is onto. If \( f \) satisfies part (1)(a) of Lemma 9, then

\[
f\Lambda_1'(s_1, x_1) = f\Lambda_1'(s_1) = \Lambda_2' f(s_1)
\]

\[
= \Lambda_2' (f(s_1), x_2) = \Lambda_2' (f(s_1), g(x_1))
\]

for all \( s_1 \in S_1 \). By part (1)(a) of Lemma 8, this shows

\[
F_1 \cong F_2.
\]

(b) Proof similar to (a) above.

(c) Proof similar to (a) above.

(2) Suppose the O.F.M. representations of \( F_1 \) and \( F_2 \) are those in the proof of (a) above. The O.F.M. representation for \( (S_1 \otimes S_2, (x_1, x_2), \Lambda) \) is

\( (S_1 \otimes S_2, \{(x_1, x_2)\}, \Lambda') \) where

\[
\Lambda'((s_1, s_2), (x_1, x_2)) = \Lambda(s_1, s_2)
\]

for all \( (s_1, s_2) \in S_1 \otimes S_2 \). Now \( F_1 \otimes F_2 = (S_1 \otimes S_2, \{(x_1, x_2)\}, \Lambda'') \) where

\[
\Lambda''((s_1, s_2), (x_1, x_2)) = (\Lambda_1'(s_1, x_1), \Lambda_2'(s_2, x_2))
\]

\[
= (\Lambda_1(s_1), \Lambda_2(s_2)) = (s_1, s_2) = \Lambda'((s_1, s_2), (x_1, x_2)).
\]
Thus
\[ F_1 \otimes F_2 \cong (S_1 \otimes S_2, (x_1, x_2), \Lambda) \]
by reflexivity.

**DEFINITION 15.**
Suppose \( M = (S, X, \Lambda) \) is an O. F. M. and \( x \in X \). Define the S. F. M. \( M_x \) by
\[ M_x = (S, x, \Lambda_x) \]
where \( \Lambda_x \) is the transformation on \( S \) induced by \( \Lambda |_{S \otimes \{ x \}} \).

It is clear that
\[ M = \sum_{x \in X} M_x. \]

**THEOREM 4.**
Suppose \( M = (S, V, \Lambda), N = (T, W, \Gamma), \) and \( P = (U, X, \Pi) \) are O. F. M.'s. Then
\[ M \otimes N \cong P \]
if and only if there exist invertible functions \( f : S \otimes T \to U \) and \( g : V \otimes W \to X \) s.t.
\[ f(\Lambda_v(s), \Gamma_w(t)) = \Pi g(v, w)f(s, t) \]
for all \( s \in S, \ t \in T, \ v \in V, \) and \( w \in W. \)

Proof:
(1) Assume \( M \otimes N \cong P \). Then there exist invertible functions \( f : S \otimes T \to U \) and \( g : V \otimes W \to X \) s.t.
\[ f(\Lambda(s, v), \Gamma(t, w)) = \Pi g(v, w)f(s, t) \]
for all \( s \in S, \ t \in T, \ v \in V, \) and \( w \in W. \) This shows that
\[ f(\Lambda_v(s), \Gamma_w(t)) = \Pi g(v, w)f(s, t) \]
for all \( s \in S, \ t \in T, \ v \in V, \) and \( w \in W. \)

(2) Assume there exist invertible functions \( f : S \otimes T \to U \) and \( g : V \otimes W \to X \) s.t.
\[ f(\Lambda_v(s), \Gamma_w(t)) = \Pi g(v, w)f(s, t) \]
for all \( s \in S, \ t \in T, \ v \in V, \) and \( w \in W. \) Then it follows
\[
f(\Lambda(s, v), \Gamma(t, w)) = \Pi(f(s, t), g(v, w))
\]
for all \( s \in S, \ t \in T, \ v \in V, \) and \( w \in W. \) This shows that
\[
M \otimes N \cong P.
\]

COROLLARY 1 to THEOREM 4.

Suppose \( M = (S, V, A), \ N = (T, W, \Gamma) \) and \( P = (U, X, \Pi) \) are O.F.M.'s. A necessary condition that
\[
M \otimes N \cong P
\]
is that there exists an invertible function \( g: V \otimes W \rightarrow X \) s.t.
\[
M_v \otimes N_w \cong P_{g(v, w)}
\]
for all \( v \in V \) and \( w \in W. \)

Proof:
The function \( f: S \otimes T \rightarrow U \) in the statement of Theorem 4 satisfies part (1)(c) of Lemma 9 for isomorphism and the result follows directly.

COROLLARY 2 to THEOREM 4.

Let \( M = (S, V, A), \ N = (T, W, \Gamma), \) and \( P = (U, X, \Pi) \) be O.F.M.'s. Let \( g: V \otimes W \rightarrow X \) be an invertible function s.t.
\[
M_v \otimes N_w \cong P_{g(v, w)}
\]
for all \( v \in V \) and \( w \in W. \) Define \( F_{vw} = \{ f : S \otimes T \rightarrow U \text{ and } f \text{ satisfies part (1)(c) of Lemma 9 for the isomorphism of } M_v \otimes N_w \text{ and } P_{g(v, w)} \}. \)

A sufficient condition that
\[
M \otimes N \cong P
\]
is that
\[
\bigcap_{v \in V} F_{vw} \neq \emptyset
\]
\[
w \in W
\]
Proof: Suppose

\[ \forall v \in V, \quad \forall w \in W, \quad F_{vw} \neq \emptyset, \quad \text{and} \quad f \in F_{vw} \]

Then \( f \) and \( g \) satisfy the conditions of Theorem 4 and it follows that \( M \otimes N \cong P \).

Theorem 4 is due in a somewhat different form to Yoeli. This theorem and its corollaries suggest that the S.F.M. and its decomposition may play an important role in the decomposition of the F.S.M.

In section 5, attention is focused on the S.F.M. and some of the problems associated with its decomposition.

5. THE DECOMPOSITION OF THE SINGLE INPUT, OUTPUT FREE FINITE STATE MACHINE

5.1 The Transformation Finite State Machine

Suppose \( F_1 = (S_1, x_1, \Lambda_1) \) and \( F_2 = (S_2, x_2, \Lambda_2) \) are S.F.M. 's. When it is desired to consider \( F_1 + F_2 \) with \( S_1 = S_2 \), then it is necessary that \( x_1 \neq x_2 \) in order for \( F_1 + F_2 \) to be defined. In this sense the letters \( x_1 \) and \( x_2 \) play an important role in the description of \( F_1 \) and \( F_2 \). In this section, however, only the properties of S.F.M. 's due to their transformations on their state alphabets will be considered. To facilitate the discussion, the transformation F.S.M. will be defined.

DEFINITION 16.

A transformation F.S.M. (hereinafter abbreviated T.F.M.) is a S.F.M. of the type

\[ (S, 1, \Lambda). \]

Clearly the double

\[ (S, \Lambda) \]

contains enough information to reconstruct the triple description of the T.F.M., and accordingly a pair of the above nature will be used to specify a T.F.M.
LEMMA 10.

Suppose \( F_1 = (S_1, \Lambda_1) \) and \( F_2 = (S_2, \Lambda_2) \) are T. F. M. 's.

1. \( F_1 \otimes F_2 \cong (S_1 \otimes S_2, \Lambda) \) where \( \Lambda : S_1 \otimes S_2 \rightarrow S_1 \otimes S_2 \) is defined by

\[
\Lambda(s_1, s_2) = (\Lambda_1(s_1), \Lambda_2(s_2))
\]

for all \( s_1 \in S_1 \) and \( s_2 \in S_2 \).

2. \( F_1 + F_2 \) is defined if and only if \( S_1 \cup S_2 \) is an alphabet and

\[
S_1 \cap S_2 = \Phi.
\]

If defined, \( F_1 + F_2 = (S_1 \cup S_2, \Lambda) \) where \( \Lambda : S_1 \cup S_2 \rightarrow S_1 \cup S_2 \) is defined by

\[
\Lambda(s) = \begin{cases} 
\Lambda_1(s) & \text{if } s \in S_1 \\
\Lambda_2(s) & \text{if } s \in S_2.
\end{cases}
\]

Proof:
The proof is not difficult and is omitted.

DEFINITION 17.

Let \( f : A \rightarrow A \) be a transformation on the set \( A \). Suppose \( n \geq 0 \).

1. Define \( f^n : A \rightarrow A \) recursively by

\[
f^0(a) = a
\]

and

\[
f^n(a) = f^{n-1}(a)
\]

for all \( a \in A \).

2. Define \( \hat{f} : 2^A \rightarrow 2^A \) by

\[
\hat{f}(A') = \{ a | \text{a} = f(a') \text{ and } a' \in A' \}
\]

for all \( A' \subseteq A \).

3. Define \( \hat{f}^{-1} : 2^A \rightarrow 2^A \) by

\[
\hat{f}^{-1}(A') = \{ a' | \text{a} = f(a) \text{ and } a' \in A' \}
\]

for all \( A' \subseteq A \).

4. Define \( \hat{f}^{-n} : 2^A \rightarrow 2^A \) by

\[
\hat{f}^{-n} = (\hat{f}^{-1})^n.
\]
DEFINITION 18.
Suppose $F = (S, \Lambda)$ is a T. F. M. and $s \in S$.
(1) $s$ is cyclic if
$$E_n (n \geq 1 \text{ and } \Lambda^n(s) = s).$$
(2) $s$ is maximal if
$$\Lambda^{-1}(s) = \emptyset.$$
(3) Suppose
$$E_n (n > 1, \Lambda^n(s) = s, \text{ and } \Lambda^i(s) \neq \Lambda^j(s) \text{ for } 1 \leq i < j \leq n-1).$$
Then
$$\{\Lambda^i(s) \mid 0 \leq i \leq n-1\}$$
is called a cycle of period $n$.
(4) Suppose $s$ is maximal, $m \geq 1$, $\Lambda^m(s)$ is not cyclic, and $\Lambda^m(s)$ is cyclic. Then
$$\{\Lambda^i(s) \mid 0 \leq i \leq m-1\}$$
is called the maximal chain of length $m$ associated with $s$.
(5) For $F$ the following subsets of $S$ are defined:
$$S^c = \{s \mid s \in S \text{ and } s \text{ is cyclic }\}$$
$$S^m = \{s \mid s \in S \text{ and } s \text{ is maximal }\}$$
and
$$S^t = \{s \mid s \in S \text{ and } s's \text{ maximal chain is as long or longer than any other maximal chain in } F\}.$$
(4) If \( C \) and \( C' \) are cycles, then either
\[ C = C' \text{ or } C \cap C' = \emptyset. \]

(5) If \( M \) is a maximal chain and \( s \in M \), then \( s \) is not cyclic.

(6) If \( s \) is cyclic, then
\[ Ec (C \text{ is a cycle and } s \in C). \]

(7) If \( s \) is not cyclic, then
\[ Em (M \text{ is a maximal chain and } s \in M). \]

Proof:

(1) Suppose \( |S| = r \). Consider the sequence \( s, A(s), \ldots, A^r(s) \). This sequence contains \( r + 1 \) states, but since there are only \( r \) different states,
\[ E(i, j) \text{ for } 0 \leq i < j \leq r \text{ and } A^i(s) = A^j(s). \]

This shows that \( A^i(s) \) is cyclic.

(2) If \( s \) is cyclic, then
\[ Em (m \geq 1 \text{ and } A^m(s) = s). \]

This shows that for all \( n \geq 0 \)
\[ A^n(s) = A^m A^n(s) \]
and by associativity that
\[ A^m A^n(s) = A^n(s). \]

(3) If \( C \) is a cycle
\[ Es' (s' \in S, A^n(s') = s', \text{ and } C = \{A^i(s') \mid 0 \leq i \leq n-1 \}). \]

Thus \( s \in C \) implies
\[ Ej (0 \leq j \leq n-1 \text{ and } s = A^j(s')). \]
By Definition 18(1), $s'$ is cyclic and it follows from (2) above that $s$ is cyclic.

(4) Suppose $C$ is a cycle of period $n$, $s \in S$, and

$$C = \{ \Lambda_i^1(s) \mid 0 \leq i \leq n-1 \}.$$  

Then if $0 \leq k \leq n-1$

$$C = \{ \Lambda_i^1 \Lambda_k(s) \mid 0 \leq i \leq n-1 \}.$$

This follows from the fact that $\Lambda_n^i(s) = s$ so that

$$C = \{ \Lambda_i^1(s) \mid 0 \leq i \leq n-k-1 \text{ or } n-k \leq i \leq n-1 \}.$$

$$= \{ \Lambda_i^1 \Lambda_k(s) \mid 0 \leq i \leq n-1 \}.$$

Now suppose $C'$ is a cycle of period $m$. Suppose also that $C \cap C' \neq \emptyset$ and that $s \in C \cap C'$.

Then

$$C = \{ \Lambda_i^1(s) \mid 0 \leq i \leq n-1 \}$$

and

$$C' = \{ \Lambda_i^1(s) \mid 0 \leq j \leq m-1 \}.$$

Suppose $m \geq n$ then

$$\Lambda_m^i(s) = \Lambda_n^i(s) = s$$

and it follows that $n = m$, since otherwise $C'$ could not be a cycle. This shows that

$$C' = C.$$

(5) If $M$ is a maximal chain, then $E(s', n)$ ($s' \in S$, $n \geq 1$, $s'$ is maximal).

$\Lambda^{n-1}(s')$ is not cyclic, $\Lambda^n(s')$ is cyclic, and $M = \{ \Lambda_i^1(s') \mid 0 \leq i \leq n-1 \}$.

Thus $s \in M$ implies

$$E \{ 0 \leq j \leq n-1 \text{ and } s = \Lambda_j^1(s') \}.$$

By (2) above and the fact that $\Lambda^{n-1}(s')$ is not cyclic, it follows that $s$ is not cyclic.
(6) If \( s \) is cyclic,

\[ \text{En} \ (n \geq 1 \text{ and } \Lambda^n(s) = s). \]

Let \( m \) be the smallest such \( n \). \( \Lambda^m(s) = s \). Then for all \( i \) and \( j \) s.t.
\[ 0 \leq i < j \leq m-1, \]
\[ \Lambda^i(s) \neq \Lambda^j(s). \]

Otherwise

\[ E(i, j) \ (\Lambda^i(s) = \Lambda^j(s) \text{ and } 0 \leq i < j \leq m-1). \]

This implies

\[ \Lambda^{j+m-1}(s) = \Lambda^m(s). \]

implies

\[ \Lambda^{j-i}(s) = \Lambda^m(s) = s, \]

which is a contradiction, since \( 1 \leq j-i < m \), and \( m \) was chosen to be the smallest positive integer s.t. \( \Lambda^m(s) = s \). It follows that

\[ \{ \Lambda^i(s) | 0 \leq i \leq m-1 \} \]

is a cycle.

(7) Suppose \( s \) is not cyclic. Suppose also that \#(S) = n. Consider the sequence
\( s, \Lambda^{-1}(s), \ldots, \Lambda^{-n}(s). \) If \( s_1 \in \Lambda^{-1}(s) \), then by (2) above, \( s_1 \) is not cyclic since
\( s = \Lambda^1(s_1) \). It follows that

\[ \Lambda^{-i}(s) \cap \Lambda^{-j}(s) = \emptyset, \quad 1 \leq i < j \leq n \]

Since \#(S) = n, and the elements of the \( n+1 \) sets \( s, \Lambda^{-1}(s), \ldots, \Lambda^{-n}(s) \) are disjoint, it follows that

\[ \Lambda^{-n}(s) = \emptyset. \]

* \( s \) will be used in place of the set \( \{ s \} \).
Let \( k \) be the largest non-negative integer s.t. \( \Lambda^{-k}(s) \neq \emptyset \). Suppose \( s_k \in \Lambda^{-k}(s) \).
Then \( s_k \) is maximal and
\[
\Lambda^{k}(s_k) = s.
\]

Let \( m \) be the least positive integer s.t. \( \Lambda^{m}(s_k) \) is cyclic. Then \( m > k \) and
\[
s \in \{ \Lambda^{i}(s_k) \mid 0 \leq i \leq m - 1 \}.
\]

It is clear that \( \{ \Lambda^{i}(s_k) \mid 0 \leq i \leq m - 1 \} \) is a maximal chain.

**Corollary to Lemma 11.**
Suppose \( F = (S, \Lambda) \) is a T. F. M. whose cycles are \( C_1, C_2, \ldots, C_m \) and whose maximal chains are \( M_1, M_2, \ldots, M_n \). Then
\[
\left( \bigcup_{1 \leq i \leq m} C_i \right) \cup \left( \bigcup_{1 \leq j \leq n} M_j \right) = S
\]
and
\[
C_i \cap M_j = \emptyset, \quad 1 \leq i \leq m, 1 \leq j \leq n.
\]

**Proof:**
The proof follows directly from Lemmas 11(3), 11(5), 11(6), and 11(7).

**Lemma 12.**
Suppose \( F_1 = (S_1, \Lambda_1) \) and \( F_2 = (S_2, \Lambda_2) \) are T. F. M.'s and \( n \geq 0 \). Suppose also that \( f : S_1 \rightarrow S_2 \) has the property that
\[
f\Lambda_1(s_1) = \Lambda_2 f(s_1)
\]
for all \( s_1 \in S_1 \). Then
\[
f\Lambda_1^{n}(s_1) = \Lambda_2^{n} f(s_1)
\]
for all \( s_1 \in S_1 \).

**Proof:**
The proof is by induction on the exponents of \( \Lambda_1 \) and \( \Lambda_2 \), and is divided into two parts.
(1) For \( n = 0 \), the equality holds, since
\[
f\Lambda_1^{0}(s_1) = f(s_1) = \Lambda_2^{0} f(s_1)
\]
for all \( s_1 \in S_1 \).
(2) Assume \( n \geq 1 \) and the equality holds for \( n - 1 \). Then

\[
A^n f(s_1) = A^{n-1} f(s_1) = A^n f(s_1) = A^{n-1} f(s_1)
\]

for all \( s_1 \in S_1 \).

5.2 The \( \Lambda \) and \( \Lambda^{-1} \) Transformation Finite State Machines

LEMMA 13.

Suppose \( F = (S, \Lambda) \) is a T. F. M. whose longest maximal chain is of length \( n \).

1. \( S \supseteq \Lambda(S) \supseteq \cdots \supseteq \Lambda^n(S) = \Lambda^{n+1}(S) = S^c \)
2. \( S^c \subseteq \Lambda^{-1}(S^c) \subseteq \cdots \subseteq \Lambda^{-(n+1)}(S^c) = \Lambda^{-n}(S^c) = S \)
3. \( \Lambda(S) = S - S^m \)
4. \( \Lambda^{-(n-1)}(S^c) = S - S^l \)
5. \( \#(S - \Lambda(S)) \geq \#(\Lambda(S) - \Lambda^2(S)) \geq \ldots \geq \)

Proof:

(1) The proof is by induction on \( n \).

(a) If \( n = 0 \), then \( F \) has no maximal chains. By Lemma 11(7) it follows that \( F \) has no non-cyclic states. Hence

\[
\Lambda^0(S) = S = S^c.
\]

Suppose \( s \in S^c \). Then

\[
Em (m \geq 1 \text{ and } \Lambda^m(s) = s).
\]

By Lemma 11(2)

\[
\Lambda^{m-1}(s) \in S^c.
\]

It follows that

\[
\Lambda(S) = \Lambda(S^c) = S^c = \Lambda^0(S).
\]

(b) Suppose \( n \geq 1 \) and (1) holds for \( n - 1 \). Since \( n \geq 1 \), it follows that there is at least one maximal state \( s' \in S \). Suppose that the maximal chain associated with \( s' \) has length \( n \).
Since $s'$ is maximal
\[ \hat{\Lambda}^{-1}(s') = \emptyset, \]
and hence
\[ s' \notin \hat{\Lambda}(S). \]

It follows that
\[ \hat{\Lambda}(S) \subseteq S. \]

Define $\Lambda_1 : \hat{\Lambda}(S) \rightarrow \hat{\Lambda}(S)$ by
\[ \Lambda_1(s) = \Lambda(s) \]
for all $s \in \hat{\Lambda}(S)$. Clearly $F_1 = (\hat{\Lambda}(S), \Lambda_1)$ is a T. F. M., since it is clear that
\[ \hat{\Lambda}_1 \hat{\Lambda}(S) = \hat{\Lambda}^2(S) \subseteq \hat{\Lambda}(S). \]

Now consider the maximal chain
\[ \{ \Lambda_i(s') \mid 0 \leq i \leq n-1 \} \]
in $F$. It was shown that $s' \notin \hat{\Lambda}(S)$. Note that
\[ \{ \Lambda_i(s') \mid 1 \leq i \leq n-1 \} \]
is a maximal chain of length $n - 1$ in $F_1$. It follows that $F_1$ has no maximal chains of length $n$ and that its longest maximal chain is of length $n - 1$. By assumption, (1) above holds for $F_1$, and
\[ \hat{\Lambda}(S) \supseteq \hat{\Lambda}_1 \hat{\Lambda}(S) \supseteq \ldots \supseteq \hat{\Lambda}_1^{n-1} \hat{\Lambda}(S) = \hat{\Lambda}_1^n \hat{\Lambda}(S) = \hat{\Lambda}(S)^c. \]

This, $S \supseteq \hat{\Lambda}(S)$, and $\hat{\Lambda}(S^c) = S^c$ imply
\[ S \supseteq \hat{\Lambda}(S) \supseteq \ldots \supseteq \hat{\Lambda}^n(S) = \hat{\Lambda}^{n+1}(S) = S^c. \]

(2) Suppose $s \in S$, $s$ is maximal, and the maximal chain associated with $s$ has length $n$. Then
\[ \Lambda^n(s) \in S^c \]
and

\[ \Lambda^{n-1}(s) \notin S^c. \]

This implies

(a) \[ \Lambda^i(s) \in \hat{\Lambda}^{-(n-i)}(S^c), \quad 0 \leq i \leq n-1 \]

but

(b) \[ \Lambda^i(s) \notin \hat{\Lambda}^{-(n-(i+1))}(S^c), \quad 0 \leq i \leq n-1 \]

Now suppose \( s' \in S \). Then, since the longest maximal chain in \( F \) is of length \( n \), it follows that

\[ s' \in \hat{\Lambda}^{-n}(S^c). \]

This shows that

(c) \[ S = \hat{\Lambda}^{-n}(S^c). \]

It is not difficult to see that

\[ S^c \subseteq \hat{\Lambda}^{-1}(S^c) \subseteq \hat{\Lambda}^{-2}(S^c) \subseteq \ldots, \]

and it follows from (a), (b), and (c) above that

\[ S^c \subseteq \hat{\Lambda}^{-1}(S^c) \subseteq \hat{\Lambda}^{-2}(S^c) \subseteq \ldots \subseteq \hat{\Lambda}^{-n}(S^c) = \hat{\Lambda}^{-(n+1)}(S^c) = S. \]

(3) If \( s \in S^m \), then

\[ \hat{\Lambda}^{-1}(s) = \emptyset \]

and it follows that

\[ s \notin \hat{\Lambda}(S). \]

Conversely if \( s \notin \hat{\Lambda}(S) \), then

\[ \hat{\Lambda}^{-1}(s) = \emptyset. \]
and hence
\[ s \in S^m. \]

It follows that
\[ \Lambda(S) = S - S^m. \]

(4) Suppose \( s \in S^t. \) Then
\[ \Lambda^{n-1}(s) \notin S^c \]
and it follows that
\[ s \notin \Lambda^{-(n-1)}(S^c). \]
Conversely suppose \( s \notin \Lambda^{-(n-1)}(S^c). \) Then by (2) above
\[ s \notin S^c. \]

By Lemma 11(7)
\[ EM \text{ (M is a maximal chain and } s \in M). \]

Suppose M's maximal state is \( s' \) and its length is \( m \leq n. \) Then
\[ s = \Lambda^i(s') \]
for some \( i \) s.t. \( 0 \leq i \leq m-1. \) Now
\[ \Lambda^{n-1}(s) = \Lambda^{n+i-1}(s') \notin S^c \]
implies
\[ m - 1 \geq n + i - 1, \]
but this and
\[ m \leq n \text{ and } i \geq 0 \]
imply
\[ m = n \text{ and } i = 0. \]
Thus
\[ s = s', \]
and it follows that
\[ s \in S_t. \]

(5) The range of \( \Lambda|_{S - \Lambda(S)} \) is \( \Lambda(S - \Lambda(S)) \). Since a function is onto its range, it follows from Lemma 3(1) that

\[ \text{(a)} \quad \#(S - \Lambda(S)) \geq \#(\Lambda(S - \Lambda(S))). \]

Suppose \( s \in \Lambda(S) - \Lambda^2(S) \). Assume \( \Lambda^{-1}(s) \subseteq \Lambda(S) \). Then \( s \in \Lambda^2(S) \), a contradiction since \( s \in \Lambda(S) - \Lambda^2(S) \). It follows that

\[ \exists s' \quad (\Lambda(s') = s, \ s' \in S, \text{ and } s' \not\in \Lambda(S)). \]

That is,
\[ s' \in S - \Lambda(S). \]

This implies
\[ s \in \Lambda(S - \Lambda(S)), \]
and it follows that
\[ \Lambda(S) - \Lambda^2(S) \subseteq \Lambda(S - \Lambda(S)). \]

This and (a) above imply
\[ \#(S - \Lambda(S)) \geq \#(\Lambda(S) - \Lambda^2(S)). \]

The general result follows by recursion.

DEFINITION 19.
Suppose \( F = (S, \Lambda) \) is a T. F. M. and \( n \geq 0 \).
(1) Define the T. F. M. $\Lambda^nF$ by

$$\Lambda^nF = (\hat{\Lambda}^n(S), \Gamma)$$

where $\Gamma : \hat{\Lambda}^n(S) \rightarrow \hat{\Lambda}^n(S)$ is defined by

$$\Gamma(s) = \Lambda(s)$$

for all $s \in \hat{\Lambda}^n(S)$.

(2) Define the T. F. M. $F_\infty$ by

$$F_\infty = (S^C, \Pi)$$

where $\Pi : S^C \rightarrow S^C$ is defined by

$$\Pi(s) = \Lambda(s)$$

for all $s \in S^C$.

(3) Define the T. F. M. $\Lambda^{-n}F_\infty$ by

$$\Lambda^{-n}F_\infty = (\hat{\Lambda}^{-n}(S^C), \Theta)$$

where $\Theta : \hat{\Lambda}^{-n}(S^C) \rightarrow \hat{\Lambda}^{-n}(S^C)$ is defined by

$$\Theta(s) = \Lambda(s)$$

for all $s \in \hat{\Lambda}^{-n}(S^C)$.

In the terminology of Definition 19, Lemma 13(1) guarantees that

$$\hat{\Gamma} \Lambda^n(S) \subseteq \hat{\Lambda}^n(S)$$

and that

$$\hat{\Pi}(S^C) = S^C.$$

Lemma 13(2) guarantees that $\hat{\Lambda}^{-n-1}(S^C) \subseteq \hat{\Lambda}^{-n}(S^C)$, which in turn guarantees that

$$\hat{\Theta} \hat{\Lambda}^{-n}(S^C) \subseteq \hat{\Lambda}^{-n}(S^C).$$

This shows that $\Lambda^nF$, $F_\infty$, and $\Lambda^{-n}F_\infty$ are indeed all T. F. M.'s as claimed.
EXAMPLE 10.

In Figure 9, the T. F. M. $F = (S, \Lambda)$ is shown. Shown with $F$ are $\Lambda F$, $\Lambda^2 F$, $F_\infty$, $\Lambda^{-1} F_\infty$, and $\Lambda^{-2} F_\infty$. Note that $\Lambda^n F = \Lambda^2 F = F_\infty$ for $n \geq 2$, and that $\Lambda^n F_\infty = \Lambda^{-2} F_\infty = F$ for $n \geq 2$.

Figure 9. $F$ is a T. F. M. $(S, \Lambda)$, where $\Lambda$ is defined by the State Diagram of $F$. Shown also are $F_\infty$, $\Lambda F$, $\Lambda^2 F$, $\Lambda^{-1} F_\infty$, and $\Lambda^{-2} F_\infty$.

THEOREM 5.

Suppose $F = (S, \Lambda)$ and $G = (T, \Gamma)$ are T. F. M.'s. Suppose also that $F$'s longest maximal chain has length $p$, $G$'s longest maximal chain has length $q$, and $n \geq 0$. 

(1) $\mathcal{F} \supseteq \mathcal{G}$ implies:
   (a) $\Lambda^n \mathcal{F} \supseteq \Gamma^n \mathcal{G}$
   (b) $\mathcal{F}_\infty \supseteq \mathcal{G}_\infty$
   (c) $p \geq q$

(2) $\mathcal{F} \subseteq \mathcal{G}$ implies:
   (a) $\Lambda^n \mathcal{F} \subseteq \Gamma^n \mathcal{G}$
   (b) $\mathcal{F}_\infty \subseteq \mathcal{G}_\infty$
   (c) $p \leq q$
   (d) $\Lambda^-n \mathcal{F}_\infty \subseteq \Gamma^-n \mathcal{G}_\infty$

(3) $\mathcal{F} \cong \mathcal{G}$ implies:
   (a) $\Lambda^n \mathcal{F} \cong \Gamma^n \mathcal{G}$
   (b) $\mathcal{F}_\infty \cong \mathcal{G}_\infty$
   (c) $p = q$
   (d) $\Lambda^-n \mathcal{F}_\infty \cong \Gamma^-n \mathcal{G}_\infty$

Proof:
Suppose $f : \mathcal{S} \rightarrow \mathcal{T}$ s.t.

(i) $f \Lambda (s) = \Gamma f(s)$

for all $s \in \mathcal{S}$. Then for any $\mathcal{S}' \subseteq \mathcal{S}$ s.t. $\Lambda (s') \subseteq s'$

$f|_{\mathcal{S}'} \Lambda (s) = \Gamma f|_{\mathcal{S}'}(s)$

for all $s \in \mathcal{S}'$. Now suppose $s \in \Lambda^n (\mathcal{S})$. Then

$E \mathcal{S}' (s' \in \mathcal{S}$ and $\Lambda^n (s') = s)$.

By (i), and Lemma 12,

$f(s) = f \Lambda^n (s') = \Gamma^n f(s')$.

It follows that if $g_n = f \Lambda^n (\mathcal{S})$, then

(ii) $\hat{g}_n \Lambda^n (\mathcal{S}) \subseteq \Gamma^n (\mathcal{T})$. 
By Lemma 13(1), if \( m = \max(p, q) \), then

(iii) \( \hat{\Lambda}^m(S) = S^c \) and \( \hat{\Gamma}^m(T) = T^c \).

From (ii), it follows that

(iv) \( \hat{g}_m(S^c) \subseteq T^c \).

Now suppose \( s \in \hat{\Lambda}^{-n}(S^c) \). Then

\[ \Lambda^n(s) \in S^c. \]

By (iii), and Lemma 12

\[ \Gamma^n f(s) = f \Lambda^n(s) = g_m \Lambda^n(s) \in T^c. \]

It follows that if \( h_n = f |_{\hat{\Lambda}^{-n}(S^c)} \), then

(v) \( \hat{h}_n \hat{\Lambda}^{-n}(S^c) \subseteq \hat{\Gamma}^{-n}(T^c) \).

Expressions (ii), (iv) and (v) show that \( \hat{g}_n : \hat{\Lambda}^n(S) \to \Gamma^n(T) \), \( g_m : S^c \to T^c \) and \( h_n : \hat{\Lambda}^{-n}(S^c) \to \hat{\Gamma}^{-n}(T^c) \) are well defined.

1. Assume \( f \) is onto.
   a. If \( t' \in \Gamma^n(T) \), then
      \[ Et (t \in T \text{ and } \Gamma^n(t) = t'). \]

Since \( f \) is onto

\[ Es (s \in S \text{ and } f(s) = t). \]

By Lemma 12

\[ f \Lambda^n(s) = \Gamma^n f(s) = t. \]

It follows that \( g_n \) is onto, and hence that

\[ \Lambda^n F \geq \Gamma^n G. \]

b. By (a) above, (iii), and Definition 19

\[ F_\infty = \Lambda^m F \geq \Gamma^m G = G_\infty. \]
(c) Assume \( q \geq 1 \). (If \( q = 0 \) the result is trivial.) Assume \( q > p \). Then \( q-1 \geq p \), and by Lemma 13(1)

\[ \hat{\Lambda}^{q-1}(S) = S^c \]

and

\[ \hat{\Gamma}^{q-1}(T) = T^c. \]

By (iv)

\[ \hat{\epsilon}^{q-1}(S) = \hat{\epsilon}^{m}(S) \subseteq T^c \subseteq \hat{\Gamma}^{q-1}(T). \]

But by the proof of (a) above, it follows that \( \epsilon^{q-1} \) is onto \( \hat{\Gamma}^{q-1}(T) \), a contradiction. Thus

\[ p \geq q. \]

(2) If \( f \) is one-one,

(a) It is obvious that \( g_n \) is one-one, and it follows that

\[ \Lambda^n F \subseteq \Gamma^n G. \]

(b) By (a) above, (iii), and Definition 19,

\[ F_\infty = \Lambda^m F \subseteq \Gamma^m G = G_\infty. \]

(c) Assume \( p \geq 1 \). (If \( p = 0 \), the result is trivial.) Assume \( p > q \). Then \( p-1 \geq q \), and by Lemma 13(1)

\[ \hat{\Lambda}^{p-1}(S) \supset S^c \]

and

\[ \hat{\Gamma}^{p-1}(T) = T^c. \]

Suppose \( s \in \hat{\Lambda}^{p-1}(S) \setminus S^c \). Then by (ii)

\[ g_{p-1}(s) \in T^c. \]

Thus

\[ \forall r \left( \Gamma^r g_{p-1}(s) = g_{p-1}(s) \right). \]
By Lemma 12

\[ g_{p-1} \Lambda^r(s) = \Gamma^r g_{p-1} (s) = g_{p-1} (s). \]

Since \( g_{p-1} \) is one-one, it follows that

\[ \Lambda^r(s) = s, \]

a contradiction, since \( s \) was chosen to be non-cyclic. This shows

\[ p \leq q. \]

(d) If \( f \) is one-one, it is obvious that \( h_n \) is one-one, and it follows that

\[ \Lambda^{-n} F_{\omega} \subseteq \Gamma^{-n} G_{\omega}. \]

(3) The proof follows from (b) above and Lemma 4(7).

**COROLLARY TO THEOREM 5.**

Suppose \( F = (S, \Lambda) \) and \( G = (T, \Gamma) \) are \( T.F.M.'s, \) \( F \geq G, \) and \( f : S \rightarrow T \) satisfies the conditions of Lemma 9 (i) for homorphism. Then

\[ \hat{f}(S^c) \subseteq T^c \]

and

\[ \hat{f}(S^m) \supseteq T^m. \]

**Proof:**

Show that \( \hat{f}(S^c) \subseteq T^c \) follows immediately from (iv) of Theorem 5. By Lemma 13(3)

\[ S^m \cup \hat{\Lambda}(S) = S \]

and

\[ T^m \cap \hat{f}(T) = \emptyset. \]

It follows by the use of (ii) of Theorem 5 that

\[ \hat{\Lambda}(S) \cap T^m = \emptyset. \]
Since \( f \) is onto,

\[ \hat{f}(S) = f(S) \cup f(A(S)) \supseteq T^m. \]

Intersecting \( T^m \) with both sides of the expression above gives

\[ T^m \cap \hat{f}(S^m) \supseteq T^m, \]

which shows

\[ \hat{f}(S^m) \supseteq T^m. \]

**LEMMA 14.**

Suppose \( F_1 = (S_1, A_1) \) and \( F_2 = (S_2, A_2) \) are S. F. M.'s, and \( n \geq 0 \).

1. \( F = F_1 \otimes F_2 \) implies
   (a) \( \Lambda^n F = \Lambda^n F_1 \otimes \Lambda^n F_2 \)
   (b) \( F_0 = F_1 \otimes F_2 \)
   (c) \( \Lambda^{-n} F_0 = \Lambda^{-n} F_1 \otimes \Lambda^{-n} F_2 \)

2. If \( F_1 + F_2 \) is defined, then \( F = F_1 + F_2 \) implies
   (a) \( \Lambda^n F = \Lambda^n F_1 + \Lambda^n F_2 \)
   (b) \( F_0 = F_1 + F_2 \)
   (c) \( \Lambda^{-n} F_0 = \Lambda^{-n} F_1 + \Lambda^{-n} F_2 \)

**Proof:**

1. It is merely necessary to show that
   (a) \( \Lambda^n (S_1 \otimes S_2) = \Lambda^n (S_1) \otimes \Lambda^n (S_2) \).

Now

\[ \Lambda^n (S_1 \otimes S_2) = \{ (\Lambda^n (s_1), \Lambda^n (s_2)) | (s_1, s_2) \in S_1 \otimes S_2 \} \]

and this is clearly

\[ \Lambda^n (S_1) \otimes \Lambda^n (S_2). \]

(b) By (a) above, and Lemma 13(1)

\[ (S_1 \otimes S_2)^c = S_1^c \otimes S_2^c, \]

and the result follows.
(c) It is merely necessary to show that
\[ \widehat{\Lambda}^{-n}(S_1^C \otimes S_2^C) = \widehat{\Lambda}_1^{-n}(S_1^C) \otimes \widehat{\Lambda}_2^{-n}(S_2^C). \]
This is obvious.

(2) The proof is as easy as that of (a) above and is omitted.

THEOREM 6.
Suppose \( F = (S, \Lambda) \), \( G = (T, \Gamma) \), and \( H = (W, \Pi) \) are T. F. M.'s and \( n \geq 0 \). Let \( \sim \) stand for any of the relations \( \geq, \subseteq, \) or \( \succeq \).

(1) \( H \sim F \otimes G \) implies
   \[ (a) \ \Pi^n H \sim \Lambda^n F \otimes \Gamma^n G \]
   \[ (b) \ \Pi^n H \sim \Lambda^n F_\infty \otimes \Gamma^n G \]

(2) If \( F + G \) is defined, then \( H \sim F + G \) implies
   \[ (a) \ \Pi^n H \sim \Lambda^n F + \Gamma^n G \]
   \[ (b) \ \Pi^n H \sim \Lambda^n F_\infty + \Gamma^n G_\infty \]

Proof:
The proof follows from (2)(a) and (2)(d) of Theorem 5, and Lemma 14.

5.3 Non-Subtractable Transformation Finite State Machines

DEFINITION 20.
A T. F. M. \( F \) is called a non-subtractable T. F. M. (hereinafter abbreviated N. S. T. F. M.) if

\[ E(F_1, F_2) \ (F_1 \text{ and } F_2 \text{ are T. F. M.'s, } F = F_1 + F_2, \]
\[ F \neq F_1, \text{ and } F \neq F_2). \]

LEMMA 15.
Let \( F = (S, \Lambda) \) be a T. F. M. Then \( F \) is a N. S. T. F. M. if and only if \( F \) has one cycle.

Proof:
The proof will be divided into two parts.

(1) Suppose \( F \) has one cycle.
Assume

\[ E(F_1, F_2) \ (F_1 \text{ and } F_2 \text{ are T. F. M.'s, } F = F_1 + F_2, \ F \neq F_1, \text{ and } F \neq F_2). \]
Suppose $F_1 = (S_1, \Lambda_1)$ and $F_2 = (S_2, \Lambda_2)$. Since $F_1 \not\equiv F$ and $F_2 \not\equiv F$, it follows that $S_1 \not\equiv \emptyset$ and $S_2 \not\equiv \emptyset$. By Lemma 11(1)

$$S_1^C \not\equiv \emptyset \text{ and } S_2^C \not\equiv \emptyset.$$ 

Suppose $s_1 \in S_1^C$ and $s_2 \in S_2^C$. By Lemma 14(2)(b), $s_1$ and $s_2 \in S^C$. Since $F$ has only one cycle, Lemma 11(6) implies that $s_1$ and $s_2$ are in the same cycle. The proof of Lemma 11(4) implies

$$E_n (n \geq 0 \text{ and } s_2 = \Lambda^n(s_1)),$$

and this implies

$$s_2 = \Lambda^n(s_1) = \Lambda^n_1(s_1) \in S_1.$$

Thus

$$S_1 \cap S_2 \not\equiv \emptyset$$

and $F_1 + F_2$ is not well defined; this is a contradiction since it was assumed $F = F_1 + F_2$. It follows that $F$ is a N.S.T.F.M.

(2) Assume $F$ has $n$ cycles $C_1, C_2, \ldots, C_n$ where $n \geq 2$. Define the relation $\sim$ on $S$ as follows:

$$s_1 \sim s_2 \text{ if and only if } E(m, n) (\Lambda^m(s_1) \text{ and } \Lambda^n(s_2) \text{ are in the same cycle.})$$

It is not difficult to see that $\sim$ is an equivalence relation and that

$$s \sim \Lambda(s)$$

for all $s \in S$.

Suppose $c_i \in C_i$ for $1 \leq i \leq n$.

Let

$$F_i = ([c_i], \Lambda_i)^*$$

$$*[c_i] = \{ s | s \sim c_i \}$$
where \( \Lambda_1 : [c_j] \rightarrow [c_j] \) is defined by

\[ \Lambda_1(s) = \Lambda(s) \]

for all \( s \in [c_j] \). Clearly \( \Lambda_1 \) is well defined since \( s \sim \Lambda(s) \) for all \( s \in S \). It is not difficult to see that

\[ F = \sum_{i=1}^{n} F_i \]

and

\[ F \not\equiv \sum_{i \neq j} F_i \quad 1 \leq j \leq n \]

This shows that if \( F \) is a N.S.T.F.M., then \( F \) must have one cycle.

**COROLLARY TO LEMMA 15.**

If \( F \) is a T.F.M. with \( n \) cycles, then there exist exactly \( n \) N.S.T.F.M.'s \( F_1, F_2, \ldots, F_n \) s.t.

1. \[ F = \sum_{i=1}^{n} F_i \]

and

2. \[ F \not\equiv \sum_{i \neq j} F_i \quad 1 \leq j \leq n \]

**Proof:**

Part (2) of the proof of Lemma 15 shows that if \( F \) contains \( n \) cycles, there is a system of at least \( n \) N.S.T.F.M.'s satisfying parts (1) and (2) of the Corollary to Lemma 15. But since each member of such a system must include one of \( F \)'s cycles that no other member of the system includes, it follows that the system can have at most \( n \) members.

The Corollary to Lemma 15 shows that any T.F.M. may be decomposed as the sum of N.S.T.F.M.'s. It also happens that this decomposition is unique and is the one given in part (2) of the proof of Lemma 15. Thus any T.F.M. may be said to have a canonical representation in terms of N.S.T.F.M.'s. If these results are generalized to S.F.M.'s, then any O.F.M. may be said to have a canonical representation in terms of N.S.S.F.M.'s. Here the O.F.M. is first decomposed into its component S.F.M.'s, and then these are decomposed into their component N.S.S.F.M.'s.
Though the above form of canonical representation is useful, especially in the light of Theorem 3, one desirable property is missing. It is not necessarily the case that the product of two N. S. S. F. M.'s is again a N. S. S. F. M. For example, in Figure 7 the product of M_2 with itself is not a N. S. S. F. M.

**LEMMA 16.**
If F_1 = (S_1, A_1) and F_2 = (S_2, A_2) are N. S. T. F. M.'s

(1) F_1 \supseteq F_2 implies

\#(S_2^c) divides \#(S_1^c)

(2) F_1 \subseteq F_2 implies

\#(S_2^c) = \#(S_1^c)

**Proof:**
(1) Suppose f : S_1 \to S_2 satisfies Lemma 9(1)(a) for homomorphism, and that s \in S_1 and \#(S_1^c) = n. Then

\[ \Lambda_1^n(s) = s. \]

By Lemma 12

\[ f(s) = f \Lambda_1^n(s) = \Lambda_2^n f(s). \]

Suppose \#(S_2^c) = m. Then clearly

\[ m \leq n \]

because if m > n, then \( f(s) \neq \Lambda_2^n f(s) \); this is a contradiction. Now

\[ f(s) = \Lambda_2^m f(s) = \Lambda_2^{2m} f(s) = \ldots \]

and

\[ f(s) \neq \Lambda_2^r f(s) \]

if \( r \) is not divisible by \( m \). It follows that \( m \) divides \( n \).
(2) Suppose $f : S_1 \rightarrow S_2$ satisfies Lemma 9(1)(b) for inclusion. Suppose $s \in S_1^c$, \( \#(S_1^c) = n \), and \( \#(S_2^c) = m \). By the argument in the proof of (1) above, \( m \) divides \( n \). But since \( f \) is one-one and
\[
 f(s) = \lambda_2^m f(s) = f\lambda_1^m(s),
\]
it follows that \( n \) divides \( m \) and hence that
\[
 m = n.
\]
Lemma 16(1) is due to Yoeli.

THEOREM 7.
Suppose \( F = (S, \Lambda) \) is a N. S. T. F. M. with \( \#(S^c) = n \) and \( \#(S - S^c) = m \). Then
\[
 EG (G \text{ is a N. S. T. F. M. of } r \text{ states and } F \geq G)
\]
if and only if
\[
 E(p, q) (p \text{ divides } n, 0 < q < m, \text{ and } p + q = r).
\]

Proof:
The proof will be divided into two parts.

(1) Assume \( G = (T, \Gamma) \) is a N. S. T. F. M. s. t. \( F \geq G, \#(T) = r, \#(T^c) = p \), and \( \#(T - T^c) = q \). By Lemma 16(1), \( p \) divides \( n \). By the Corollary to Theorem 5, if \( f : S \rightarrow T \) satisfies the conditions of Lemma 9(1)(b) for homomorphism, then
\[
 \hat{f}(S - S^c) \supseteq T - T^c.
\]
Since \( f \) is onto, it follows from Lemma 3(1) that
\[
 q \leq m.
\]

(2) Conversely assume
\[
 E(p, q) (p \text{ divides } n, 0 \leq q \leq m, \text{ and } p + q = r).
\]

Let \( H = (C, \Pi) \) be a N. S. T. F. M., where \( C \) is a cycle of period \( p \). Suppose \( s_0 \in S^c \) and \( c_0 \in C \). Then define \( g : S^c \rightarrow C \) by
\[
 g\lambda^i(s_0) = \Pi^i(c_0), \quad 0 \leq i \leq n-1
\]
Note that
\[ g(s_0) = gA^n(s_0) = \Pi^n(c_0) = c_0. \]

Let \( F_H = (C \cup (S - S^c), \Theta) \) where \( \Theta \) is defined by
\[
\Theta(s) = \begin{cases} 
\Pi(s) & \text{if } s \in C \\
A(s) & \text{if } s \in S - S^c \text{ and } A(s) \in S - S^c \\
gA(s) & \text{if } s \in S - S^c \text{ and } A(s) \in S^c 
\end{cases}
\]

Define \( h : S \to C \cup (S - S^c) \) by
\[
h(s) = \begin{cases} 
s & \text{if } s \in S - S^c \\
g(s) & \text{if } s \in S^c 
\end{cases}
\]

The function \( h \) is a homomorphism since it is clearly onto and
\[
hA(s) = \begin{cases} 
gA(s) & \text{if } s \in S^c \\
A(s) & \text{if } s \in S - S^c \text{ and } A(s) \in S - S^c \\
gA(s) & \text{if } s \in S - S^c \text{ and } A(s) \in S^c 
\end{cases}
\]

for all \( s \in S. \)
Thus
\[ F \geq F_H \]
and \( F_H \) has \( p + m \) states. If \( m = q \), then \( F_H \) is the required machine. Otherwise, \( m > q \geq 0. \) Letting \( W = C \cup (S - S^c), \) suppose \( w \in \Theta^{-1}(W^c). \) Define \( D_w(F_H) \) to be the N.S.T.F.M. \( (W - \{ w \}, \Delta) \) where \( \Delta : (W - \{ w \}) \to (W - \{ w \}) \) is defined by
\[
\Delta(s) = \begin{cases} 
\Theta^{p+1}(s) & \text{if } s \in \Theta^{-1}(w) \\
\Theta(s) & \text{otherwise} 
\end{cases}
\]
for all \( s \in W - \{ w \}. \) Let \( f : W \to W - \{ w \} \) be defined by
\[
f(s) = \begin{cases} 
\Theta^p(s) & \text{if } s = w \\
s & \text{otherwise} 
\end{cases}
\]
Clearly \( f \) is onto and

\[
\begin{align*}
f\Theta(s) &= \begin{cases} 
\Theta^{p+1}(s) & \text{if } s \in \Theta^{-1}(w) \\
\Theta(s) & \text{if } s = w \\
\Theta(s) & \text{otherwise}
\end{cases}
\end{align*}
\]

\[
\begin{align*}
\Theta^{p+1}(s) &= \begin{cases} 
\Theta^{p+1}(s) & \text{if } s \in \Theta^{-1}(w) \\
\Theta^{p+1}(s) & \text{if } s = w \\
\Theta(s) & \text{otherwise}
\end{cases}
\end{align*}
\]

which shows that

\[
F_H \geq D_w(F_H)
\]

and by transitivity

\[
F \geq D_w(F_H)
\]

\( D_w(F_H) \) has \( p + m - 1 \) states and by the recursive application of the \( D \) operator, a N.S.T. F.M. \( G = (T, \Gamma) \) can be obtained s.t. \( F \geq G, \ #(T^c) = p \) and \( #(T - T^c) = q. \)

**THEOREM 8.**

Let \( F_1 = (S_1, \Lambda_1) \) and \( F_2 = (S_2, \Lambda_2) \) be N.S.T. F.M.'s. Suppose \( #(S_1) = p_1 \) and \( #(S_2) = p_2. \) Let \( q \) be the greatest common divisor of \( p_1 \) and \( p_2, \) and let \( r \) be the least common multiple of \( p_1 \) and \( p_2. \) Then if \( F = (S, \Lambda) \) is a T.F.M. and \( F \simeq F_1 \otimes F_2, \) \( F \) has \( q \) cycles of period \( r. \)

**Proof:**

Suppose \( f : S_1 \otimes S_2 \to S \) satisfies the conditions of Lemma 9(1)(c) for isomorphism, and that \( s \in S^c. \) Then using the Corollary to Theorem 5,

\[
f^{-1}(s) \in S_1^c \otimes S_2^c.
\]

Suppose

\[
f^{-1}(s) = (s_1, s_2);
\]

then

\[
\Lambda^F(s) = f(\Lambda_1^F(s_1), \Lambda_2^F(s_2)) = f(s_1, s_2) = s.
\]
If \( p \) is the period of \( s \)'s cycle, then \( p \) divides \( r \). On the other hand, \( p_1 \)
 divides \( p \), and \( p_2 \) divides \( p \). Thus \( p \) is a common multiple of \( p_1 \) and \( p_2 \),
 and hence \( r \) divides \( p \). It follows that

\[
p = r.
\]

Since the period of any cycle in \( F \) is \( r \), the different cycles are disjoint by
Lemma \( 11(4) \), and there are \( p_1 p_2 \) cyclic states in \( F \). It follows by a result in
number theory* that there are

\[
q = \frac{p_1 p_2}{r}
\]
such cycles.

5.4 Generating Functions

**DEFINITION 21.**

Consider the set \( Z \) with the usual operations of addition and multiplication
defined on its members. The members of \( Z \) will be used as the coefficients in a
system of polynomial-like structure. Assume that the variables of this system are
\( \{ x^0, x^1, \ldots \} \), and that addition and multiplication (denoted by \( + \) and \( \cdot \) respectively)
in this system obey the following rules for any \( a \) and \( b \) in \( A \):

\[
\begin{align*}
ax^i + bx^i &= (a+b)x^i \\
ax^i + ax^j &= a(x^i + x^j) \\
ax^i \cdot bx^j &= ab(x^i \cdot x^j) \\
x^i \cdot (x^j + x^k) &= x^i \cdot x^j + x^i \cdot x^k
\end{align*}
\]

Three different types of systems satisfying the above requirements will be
defined.

1. The system is of Type 1 if for any variables \( x^i \) and \( x^j \),

\[
x^i \cdot x^j = \begin{cases} x^i & \text{if } i=j \\ 0 & \text{otherwise.} \end{cases}
\]

2. The system is of Type 2 if for any variables \( x^i \) and \( x^j \)

\[
x^i \cdot x^j = x^{\max(i,j)}
\]

---

*See for example, Griffin*6 for a discussion of the greatest common divisor,
and the least common multiple.
(3) The system is of Type 3 if \( x^0 \) is not a variable of the system, and for any variables \( x^i \) and \( x^j \) in the system,
\[
x^i \cdot x^j = qx^r
\]
where \( q \) is the greatest common divisor of \( i \) and \( j \), and \( r \) is the least common multiple of \( i \) and \( j \).

**Lemma 17.**
Consider Definition 21. In Systems of Types 1, 2, and 3, the operations of multiplication and addition are commutative and associative.

**Proof:**
(1) Lemma 17 is obvious except for the associativity of multiplication in Systems of Type 3. This will be proved.

(2) Consider \( a_i x^i \), \( a_j x^j \), and \( a_k x^k \). Let \( F_i \), \( F_j \), and \( F_k \) be T. F. M.'s consisting of \( a_i \) cycles of period \( i \), \( a_j \) cycles of period \( j \), and \( a_k \) cycles of period \( k \), respectively. By Theorems 3 and 8, if
\[
a_i x^i \cdot a_j x^j = b_m x^m
\]
and
\[
a_j x^j \cdot a_k x^k = b_n x^n,
\]
then \( F_i \otimes F_j \) has \( b_m \) cycles of period \( m \) and \( F_j \otimes F_k \) has \( b_n \) cycles of period \( n \).
Since
\[
(F_i \otimes F_j) \otimes F_k \cong F_i \otimes (F_j \otimes F_k),
\]
it follows that
\[
b_m x^m \cdot a_k x^k = a_i x^i \cdot b_n x^n.
\]

**Lemma 18.**
Consider Definition 21. Let \( R_1 \) be a System of Type 1 whose variables are \( \{ x^0, x^1, \ldots \} \), and \( R_2 \) be a System of Type 2 whose variables are \( \{ y^0, y^1, \ldots \} \).
Suppose

\[ a(y) = a_0y^0 + a_1y^1 + \ldots \]
\[ b(y) = b_0y^0 + b_1y^1 + \ldots \]
\[ c(y) = c_0y^0 + c_1y^1 + \ldots \]
\[ A(x) = A_0x^0 + A_1x^1 + \ldots \]
\[ B(x) = B_0x^0 + B_1x^1 + \ldots \]
\[ C(x) = C_0x^0 + C_1x^1 + \ldots \]

If

\[ c(y) = a(y) \cdot b(y), \]

\[ A_i = \sum_{j=0}^{i} a_j, \quad 0 \leq i \]
\[ B_i = \sum_{j=0}^{i} b_j, \quad 0 \leq i \]

and

\[ C_i = \sum_{j=0}^{i} c_j, \quad 0 \leq i \]

then

\[ C(x) = A(x) \cdot B(x). \]

Proof:

The proof will be given by showing that for all \( p \geq 0 \) \( C_p = A_p \cdot B_p \). The proof will be by induction on \( p \).

1. If \( p = 0 \), the result is obvious.
2. Suppose \( p \geq 1 \) and the result holds for \( p - 1 \). Now

\[ C_p = C_{p-1} + c_p, \]

and

\[ c_p = b_p \sum_{i=0}^{p-1} a_i + a_p \sum_{i=0}^{p-1} b_i + a_b \]
\[ = b_p A_{p-1} + a_b B_{p-1} + a_b b_p. \]
By assumption $C_{p-1} = A_{p-1} \cdot B_{p-1}$. Thus it follows that

$$C_p = C_{p-1} + c_p$$

$$= A_{p-1}B_{p-1} + a_p B_{p-1} + A_{p-1} b_p + a_p b_p$$

$$= (A_{p-1} + a_p) (B_{p-1} + b_p)$$

$$= A_p \cdot B_p.$$ 

Polynomials from Systems 1, 2, and 3 are used as generating functions* for counting certain types of states in T. F. M.'s and more generally in S. F. M.'s. Suppose $S_0, S_1, \ldots$ is a sequence of sets and

$$a(x) = a_0 x^0 + a_1 x^1 + \ldots$$

where

$$a_i = \#(S_i)^0 \quad 0 \leq i$$

In this case, the polynomial $a(x)$ is said to be the generating function for the sequence of sets $S_0, S_1, \ldots$

**Definition 22.**

Let $F = (S, A)$ be a T. F. M. Suppose

$$a'(v) = a'_0 v^0 + a'_1 v^1 + \ldots + a'_p v^p$$

and

$$A(v) = A_0 v^0 + A_1 v^1 + \ldots + A_p v^p$$

are polynomials of Type 1;

$$a'(x) = a'_0 x^0 + a'_1 x^1 + \ldots + a'_p x^p$$

and

$$A'(x) = A'_0 x^0 + A'_1 x^1 + \ldots + A'_p x^p$$

*More familiar types of generating functions are treated in Riordan. 7
are polynomials of Type 2; and 

\[ f(y) = f_1 y^1 + f_2 y^2 + \ldots \]

is a polynomial of Type 3.

If 

\[ a_i = \#(\Lambda^{(p-1)}(S)), \]
\[ A_i = \#(\Lambda^{-1}(S^C)), \]
\[ a_0 = a_0 \text{ and } A_i = A_0, \]
\[ a'_i = a_i - a_{i-1}, \]
\[ A'_i = A_i - A_{i-1}, \]

and

\[ f_i = \# \{ \text{cycles of period } i \text{ in } F \}, \]

then \( a(v) \) is called \( F \)'s \( (p, A) \) generating function (hereinafter abbreviated \( (p, A) \) g.f.); \( A(v) \) is called \( F \)'s \( (p, A^{-1}) \) g.f.; \( a'(x) \) is called \( F \)'s \( (p, A) \) difference g.f. (hereinafter abbreviated \( (p, A) \) d.g.f.); \( A'(x) \) is called \( F \)'s \( (p, A^{-1}) \) d.g.f.; and \( f(y) \) is called \( F \)'s cyclic g.f. (hereinafter abbreviated c.g.f.).

THEOREM 9.

Suppose \( F = (S, A) \), \( G = (T, \Gamma) \), and \( H = (W, \Pi) \) are T. F. M.'s. Suppose also that \( F \)'s longest maximal chain has length \( m \), and \( G \)'s longest maximal chain has length \( n \). Let \( p = \max(m, n) \). Then:

Define \( a(v), A(v), a'(x), A'(x), \) and \( f(y) \), to be, respectively, \( F \)'s \( (p, A) \) g.f., \( F \)'s \( (p, A^{-1}) \) g.f., \( F \)'s \( (p, A) \) d.g.f., \( F \)'s \( (p, A^{-1}) \) d.g.f., and \( F \)'s c.g.f.

Define \( b(v), B(v), b'(x), B'(x), \) and \( g(y) \), to be, respectively, \( G \)'s \( (p, \Gamma) \) g.f., \( G \)'s \( (p, \Gamma^{-1}) \) g.f., \( G \)'s \( (p, \Gamma) \) d.g.f., \( G \)'s \( (p, \Gamma^{-1}) \) d.g.f., and \( G \)'s c.g.f.

Define \( c(v), C(v), c'(x), C'(x), \) and \( h(y) \), to be, respectively, \( H \)'s \( (p, \Pi) \) g.f., \( H \)'s \( (p, \Pi^{-1}) \) g.f., \( H \)'s \( (p, \Pi) \) d.g.f., \( H \)'s \( (p, \Pi^{-1}) \) d.g.f., and \( H \)'s c.g.f.

(1) If \( H \cong F \otimes G \), then \( c(v) = a(v) \cdot b(v), C(v) = A(v) \cdot B(v), c'(x) = a'(x) \cdot b'(x), C'(x) = A'(x) \cdot B'(x), \) and \( h(y) = f(y) \cdot g(y) \).

(2) If \( F + G \) is defined and \( H \cong F + G \), then \( c(v) = a(v) + b(v), C(v) = A(v) + B(v), c'(x) = a'(x) + b'(x), C'(x) = A'(x) + B'(x), \) and \( h(y) = f(y) + g(y) \).
Proof:
(1) The proof is a consequence of Theorems 6 and 8, and Lemma 18.
(2) The proof is not difficult, and hence is omitted.

EXAMPLE 11.
Suppose \( F_1 = (S_1, \Lambda_1) \) and \( F_2 = (S_2, \Lambda_2) \) are the T.F.M.'s depicted in Figure 11 (see page 93).

(1) \( F_1 \) 's \((2, \Lambda_1)\) d. g. f. is
\[
a_1(x) = x^0 + x^1 + x^2.
\]

(2) \( F_1 \) 's \((2, \Lambda_1^{-1})\) d. g. f. is
\[
A_1(x) = x^0 + x^1 + x^2.
\]

(3) \( F_2 \) 's \((2, \Lambda_2)\) d. g. f. is
\[
a_2(x) = 2x^0 + x^2.
\]

(4) \( F_2 \) 's \((2, \Lambda_2^{-1})\) d. g. f. is
\[
A_2(x) = 2x^0 + x^1.
\]

(5) \( F_1 \otimes F_2 \)'s \((2, [\Lambda_1, \Lambda_2])\) d. g. f. is
\[
2x^0 + 2x^1 + 5x^2 = a_1(x) \cdot a_2(x).
\]

(6) \( F_1 \otimes F_2 \)'s \((2, \Lambda_1^{-1}, \Lambda_2^{-1})\) d. g. f. is
\[
2x^0 + 4x^1 + 3x^2 = A_1(x) \cdot A_2(x).
\]

COROLLARY 1 TO THEOREM 9.
Let \( F, G, H, a'(x), b'(x), c'(x), A'(x), B'(x), \) and \( C'(x) \) be the same as in Theorem 9. Suppose \( H \equiv F \otimes G \).
(1) If \( F \) has \( q \) maximal states and \( G \) has \( r \) maximal states, then \( H \) has \( q \cdot \#(T) + r \cdot \#(S) - qr \) maximal states.
(2) If \( F \) has \( q \) maximal chains of length \( p \), and \( G \) has \( r \) maximal chains of length \( p \), then \( H \) has \( q \cdot \#(T) + r \cdot \#(S) - qr \) maximal chains of length \( p \), and no maximal chains whose length is greater than \( p \).
Proof:

Let

\[ d(x) = d_0 x^0 + d_1 x^1 + \ldots + d_p x^p \]

for \( d \) equal \( a', b', c', A', B', \) or \( C' \).

(1) By Lemma 13(3),

\[
\begin{align*}
 a'_p &= \#(S - \Lambda(S)) = \#(S^\infty) \\
 b'_p &= \#(T - \Gamma(T)) = \#(T^\infty)
\end{align*}
\]

and

\[
 c'_p = \#(W - \Pi(W)) = \#(W^\infty).
\]

By Theorem 9, \( c'(x) = a'(x) \cdot b'(x) \) in a System of Type 2. This implies

\[
 c'_p = a'_p \sum_{i=0}^{p} b'_i + b'_p \sum_{i=0}^{p} a'_i \cdot b'_i,
\]

and the result follows.

(2) By Lemma 13(2),

\[
 \Lambda^{-P}(S^c) = S \text{ and } \Gamma^{-P}(T^c) = T.
\]

This shows that

\[
 \Pi^{-P}(W^c) = W,
\]

which, in turn, shows that \( H \)'s longest maximal chain has length less than or equal to \( p \). Since \( p = \max(m, n) \), it follows that either \( m = p \) or \( n = p \), or both. Assume \( m = p \). By Lemma 13(4),

\[
 A'_p = \#(S - \Lambda^{-p-1}(S^c)) = \#(S^1).
\]

Now \( n \leq p \) and either

\[
 B'_p = \#(T^1), \text{ or } B'_p = 0
\]

depending upon whether \( n = p \), or \( n < p \). In either case \( B'_p \) is the number of maximal chains of length \( p \) in \( G \), and the result follows from

\[
 C'_p = B'_p \cdot \#(S) + A'_p \cdot \#(T) - A'_p B'_p.
\]
COROLLARY 2 TO THEOREM 9.
Suppose $H = (W, \Pi)$ is a T. F. M. where the longest maximal chain has length $p$, the $(p, \Lambda)$ g. f. is
\[ c_0 v^0 + c_1 v^1 + \ldots + c_p v^p, \]
and the $(p, \Lambda^{-1})$ g. f. is
\[ C_0 v^0 + C_1 v^1 + \ldots + C_p v^p. \]
If
\[ E (F, G) \] (F and G are T. F. M.'s and $H \cong F \otimes G$),
then there exist $p$-tuples of positive integers $(a_0, a_1, \ldots, a_p)$, $(b_0, b_1, \ldots, b_p)$, $(A_0, A_1, \ldots, A_p)$, and $(B_0, B_1, \ldots, B_p)$ s.t.:

1. $a_i - b_i = c_i$ and $A_i \cdot B_i = C_i$ \hspace{1cm} $0 \leq i \leq p$
2. $a_0 \leq a_1 \leq \ldots \leq a_p$,
   $b_0 \leq b_1 \leq \ldots \leq b_p$,
   $A_0 \leq A_1 \leq \ldots \leq A_p$, and
3. $B_0 \leq B_1 \leq \ldots \leq B_p$.

1. $a_1 - a_0 \leq a_2 - a_1 \leq \ldots \leq a_p - a_{p-1}$,
   $b_1 - b_0 \leq b_2 - b_1 \leq \ldots \leq b_p - b_{p-1}$,
   $A_i - A_{i-1} = 0$ implies $A_j - A_{j-1} = 0$, \hspace{1cm} $1 \leq i \leq p$
and
1. $B_j - B_{j-1} = 0$ implies $B_j - B_{j-1} = 0$. \hspace{1cm} $1 \leq j \leq p$
Proof:
If \( F = (S, \Lambda) \) and \( G = (T, \Gamma) \) are T.F.M.'s s.t. \( H \simeq F \otimes G \), let
\( F \)'s \( (p, \Lambda) \) g.f. be
\[
a_0 v^0 + a_1 v^1 + \ldots + a_p v^p
\]
\( F \)'s \( (p, \Lambda^{-1}) \) g.f. be
\[
A_0 v^0 + A_1 v^1 + \ldots + A_p v^p
\]
\( G \)'s \( (p, \Gamma) \) g.f. be
\[
b_0 v^0 + b_1 v^1 + \ldots + b_p v^p
\]
and \( G \)'s \( (p, \Gamma^{-1}) \) g.f. be
\[
B_0 v^0 + B_1 v^1 + \ldots + B_p v^p.
\]
Clearly
\[
a_i \cdot b_i = c_i \quad \text{and} \quad A_i \cdot B_i = C_i
\]
\( 0 \leq i \leq p \)

Part (2) of Corollary 2 to Theorem 9 is implied by Lemmas 13(1) and 13(2),
and Part (3) is implied by Lemmas 13(2) and 13(5).

COROLLARY 3 TO THEOREM 9.
Let \( F = (S, \Lambda) \) and \( G = (T, \Gamma) \) be T.F.M.'s. Suppose \( p \) is the length of \( F \)'s
longest chain and \( G \) has no chains whose length is greater than \( p \). Suppose
\( F \)'s \( (p, \Lambda) \) g.f. is
\[
a_0 v^0 + a_1 v^1 + \ldots + a_p v^p,
\]
\( F \)'s \( (p, \Lambda^{-1}) \) g.f. is
\[
A_0 v^0 + A_1 v^1 + \ldots + A_p v^p,
\]
\( G \)'s \( (p, \Gamma) \) g.f. is
\[
b_0 v^0 + b_1 v^1 + \ldots + b_p v^p,
\]
and G's $(p, r^{-1})$ g.f. is
\[ B_0v^0 + B_1v^1 + \ldots + B_pv^p. \]

If
\[ \text{EH (H is a T. F. M. and G} \oplus H) \cong F), \]
then:
\[ \begin{align*}
(1) & \quad \frac{a_i}{b_i} \quad \text{and} \quad \frac{A_i}{B_i} \quad \text{are positive integers.} \\
(2) & \quad \frac{a_0}{b_0} \leq \frac{a_1}{b_1} \leq \ldots \leq \frac{a_p}{b_p} \\
\end{align*} \]

and
\[ \begin{align*}
A_0 & \leq A_1 \leq \ldots \leq A_p. \\
\text{and} & \\
\frac{a_1}{b_1} \quad \frac{a_0}{b_0} \leq \frac{a_2}{b_2} \leq \ldots \leq \frac{a_n}{b_n} \leq \frac{a_{n-1}}{b_{n-1}}. \\
\end{align*} \]

and
\[ \begin{align*}
\frac{a_1}{b_1} - \frac{a_{i-1}}{b_{i-1}} = 0 \text{ implies } \frac{a_1}{b_j} - \frac{a_{j-1}}{b_{j-1}} = 0. \\
0 \leq i \leq p \quad \text{and} \quad 0 \leq j \leq p \end{align*} \]

Proof:
If $H = (W, II)$ is a T. F. M. s.t. $F \cong G \oplus H$, let

$H$'s $(p, II)$ g.f. be
\[ c_0x^0 + c_1x^1 + \ldots + c_px^p \]

and $H$'s $(p, II^{-1})$ g.f. be
\[ C_0x^0 + C_1x^1 + \ldots + C_px^p. \]

Clearly
\[ a_i = b_i \cdot c_i \text{ and } A_i = B_i \cdot C_i. \]

0 \leq i \leq p
Thus
\[ c_i = \frac{a_i}{b_i} \quad \text{and} \quad C_i = \frac{A_i}{B_i}, \quad 0 \leq i \leq p \]
and the results follow.

THEOREM 10.

Suppose
\[ A(x) = A_0 x^0 + A_1 x^1 + \ldots + A_n x^n \]
is a polynomial from a System of Type 2, and \( A_i \neq 0 \) for \( 0 \leq i \leq n \). Then

\[ \text{EF (} F = (S, A) \text{ is a N.S.T.F.M. and } A(x) \text{ is } F's (n, \Lambda^{-1}) \text{ d.g.f.)} \]

Proof:
Consider Figure 10. Clearly \( F's (n, \Lambda^{-1}) \) d.g.f. is \( A(x) \).

COROLLARY TO THEOREM 10.

For any \( n \geq 1 \), the number of non-isomorphic N.S.T.F.M.'s having \( n \) states is bounded below by \( 2^{n-1} \).

Proof:
The proof shows that for \( n \)-state N.S.T.F.M.'s, there are \( 2^{n-1} \) different \( (n-1, \Lambda^{-1}) \) d.g.f.'s. The result then follows from Theorem 10 plus the fact that isomorphic T.F.M.'s have the same \( (n-1, \Lambda^{-1}) \) d.g.f.'s for \( 1 \leq n < \infty \). The proof is by induction on \( n \) and is divided into two parts.

1. If \( n = 1 \), there is exactly 1 = \( (2^0) \) equivalence class of N.S.T.F.M.'s having 1 state. There is just 1 \((0, \Lambda^{-1})\) d.g.f. namely \( x^0 \), and so the result holds for \( n = 1 \).

2. Suppose that \( n \geq 2 \), and that there are \( 2^{m-1} \) different \((m-1), \Lambda^{-1}\) d.g.f.'s for \( m \)-state N.S.T.F.M.'s, if \( m < n \). Clearly if \( m < n \), there are \( 2^{m-1} \) different \((n-1, \Lambda^{-1})\) d.g.f.'s for \( m \)-state N.S.T.F.M.'s.

Let
\[ A = \{ a(x) \mid a(x) \text{ is the } (n-1, \Lambda^{-1}) \text{ d.g.f. of an } m \text{-state N.S.T.F.M. for } m < m \}, \]
\[ B = A \cup \{ n x^0 \}, \]
and
\[ C = \{ c(x) \mid c(x) \text{ is the } (n-1, \Lambda^{-1}) \text{ d.g.f. of an } n \text{-state N.S.T.F.M.} \}. \]
Figure 10. A N.S.T. F.M. whose \((n, \Lambda^{-1})\) d.g.f. is \(A_0x^0 + A_1x^1 + \ldots + A_nx^n\)

For all \(b(x) \in B\) define \(f : B \to C\) by

\[
f(b(x)) = b(x) + (n-m)x^k
\]

where \(m\) is the number of states of a N.S.T. F.M. for which \(b(x)\) is the \((n-1, \Lambda^{-1})\) d.g.f. and \(k\) is the smallest positive integer for which the coefficient of \(x^k\) is 0. It is not difficult to see that \(f\) is well defined and invertible, and hence that

\[
\#(C) = \#(B) = 1 + \sum_{m=1}^{n-1} 2^{m-1} = 2^{n-1}.
\]
5.5 Decomposition of a T. F. M.

A procedure can now be given for decomposing a T. F. M. into the product of two smaller T. F. M.'s, or ascertaining that no such decomposition exists.

Suppose \( F = (S, \Lambda) \) is a T. F. M. whose longest maximal chain has length \( p \), and that \( a(x) \) is \( F \)'s \((p, \Lambda) \) g. f. , \( A(x) \) if \( F \)'s \((p, \Lambda^{-1}) \) g. f. , and \( f(y) \) is \( F \)'s c. g. f.

1. The first step is to find all sextuples

\[(b(x), c(x), B(x), C(x), g(y), h(y)) \text{ s.t.}\]

\[b(x) \cdot c(x) = a(x), \ B(x) \cdot C(x) = A(x), \text{ and } g(y) \cdot h(y) = f(y),\]

\[b(x) \not\equiv a(x) \not\equiv c(x), \ B(x) \not\equiv A(x) \not\equiv C(x), \text{ and } g(y) \not\equiv f(y) \not\equiv h(y),\]

and if

\[b(x) = b_0 x^0 + b_1 x^1 + \ldots + b_p x^p,\]
\[c(x) = c_0 x^0 + c_1 x^1 + \ldots + c_p x^p,\]
\[B(x) = B_0 x^0 + B_1 x^1 + \ldots + B_p x^p,\]

and

\[C(x) = C_0 x^0 + C_1 x^1 + \ldots + C_p x^p,\]

then

\[b_0 \leq b_1 \leq \ldots \leq b_p,\]
\[c_0 \leq c_1 \leq \ldots \leq c_p,\]
\[b_1 - b_0 \leq b_2 - b_1 \leq \ldots \leq b_p - b_{p-1},\]
\[c_1 - c_0 \leq c_2 - c_1 \leq \ldots \leq c_p - c_{p-1},\]
\[B_0 \leq B_1 \leq \ldots \leq B_p,\]
\[C_0 \leq C_1 \leq \ldots \leq C_p,\]
\[B_i - B_{i-1} = 0 \text{ implies } B_j - B_{j-1} = 0 \quad i \leq j \leq p\]

and

\[C_i - C_{i-1} = 0 \text{ implies } C_j - C_{j-1} = 0. \quad i \leq j \leq p\]
(2) If \((b(x), c(x), B(x), C(x), g(y), h(y))\) is a sextuple of the type to be found in (1) above, define T. F. M.'s \(G_0 = (T_0, \Gamma_0)\) and \(H_0 = (W_0, \Pi_0)\) s. t. \(G_0\) consists of the cycles given by \(g(y)\), and \(H_0\) consists of the cycles given by \(h(y)\). This construction is easy and it is obvious that
\[
G_0 \otimes H_0 \cong F_\infty
\]

(3) If \(G_n = (T_n, \Gamma_n)\) and \(H_n = (W_n, \Pi_n)\) are T. F. M.'s s. t. \(G_n \otimes H_n \cong \Lambda^{-n}F_\infty\) for \(0 \leq n \leq p-1\), try to find T. F. M.'s \(G_{n+1} = (T_{n+1}, \Gamma_{n+1})\) and \(H_{n+1} = (W_{n+1}, \Pi_{n+1})\) s. t. \(G_{n+1} \otimes H_{n+1} \cong \Lambda^{-(n+1)}F_\infty\), by considering only T. F. M.'s obeying the equations
\[
\begin{align*}
\Gamma_{n+1}^{-n} \Gamma_{n+1}^{-n} G_{n+1} &= G_n, \\
\Pi_{n+1}^{-n} \Pi_{n+1}^{-n} H_{n+1} &= H_n,
\end{align*}
\]
and
\[
\#(T_{n+1} - T_n) = B_{n+1} - B_n, \quad \#(W_{n+1} - W_n) = C_{n+1} - C_n.
\]

(4) If T. F. M.'s fail to exist at any point satisfying (3) above, then Theorem 6 shows that there are no T. F. M.'s \(G = (T, \Gamma)\) and \(H = (W, \Pi)\) s. t.
\[
F \cong G \otimes H
\]
and
\[
G_n \cong \Gamma^{-n}G_\infty \text{ and } H_n \cong \Pi^{-n}H_\infty.
\]

(5) If there are T. F. M.'s \(G_{n+1}\) and \(H_{n+1}\) satisfying (3) above for all \(n\) s. t. \(0 \leq n \leq p-1\), then of course
\[
F \cong G_p \otimes H_p.
\]

(6) After all T. F. M.'s \(G\) and \(H\) s. t. \(F \cong G \otimes H\) have been found for one sextuple, the process outlined above may be carried out for all remaining sextuples, thus obtaining all decompositions of \(F\), if any exist.

EXAMPLE 12.
Consider the T. F. M. \(F = (S, \Lambda)\) whose longest maximal chain has length 2 and whose \((2, \Lambda)\) g. f. is
\[
a(x) = x^0 + 25x^1 + 49x^2.
\]
The only polynomials \( b(x) \) and \( c(x) \) s.t. \( a(x) = b(x) \cdot c(x) \) and \( b(x) \neq a(x) \neq c(x) \) are
\[
b(x) = c(x) = x^0 + 5x^1 + 7x^2.
\]

Note however that
\[
5 - 1 = 4 \leq 2 = 7 - 5.
\]

It follows that such a T.F.M. is not decomposable as the product of two smaller T.F.M.'s.

**EXAMPLE 13.**

Consider the T.F.M. \( F_1 \otimes F_2 \) in Figure 11. The \((2, [\Lambda_1, \Lambda_2])^{-1} \) g.f. of \( F_1 \otimes F_2 \) is
\[
C(v) = 2v^0 + 6v^1 + 9v^2.
\]

![Diagram of T.F.M. composition](image)

**Figure 11. Example of the Composition of Two T.F.M.'s**
There is only one pair of polynomials $A(v)$ and $B(v)$ s.t. $A(v) \neq C(v) \neq B(v)$, $A(v) \cdot B(v) = C(v)$, and parts 2 and 3 of Corollary 2 to Theorem 9 are satisfied by $A(v)$ and $B(v)$. This pair is

$$A(v) = v^0 + 2v^1 + 3v^2$$

and

$$B(v) = 2v^0 + 3v^1 + 3v^2.$$ 

In this case, $F_1$ of Figure 11 is uniquely determined (up to isomorphism) by $A(v)$. The T. F. M. $F_2$ is not determined by $B(v)$ until the cycle structure has been determined. Then $F_2$ is uniquely determined by $B(v)$.

In general, it will not be the case that a T. F. M. is uniquely determined by its set of generating functions.

### 5.6 Multiple Generating Functions

The use of generating functions can be extended in several ways. Here, two such ways are suggested. Example 14 shows how for some T. F. M. $(S, \Lambda)$, the $(p, \Lambda)$ d. g. f. and $(p, \Lambda^{-1})$ d. g. f. may be combined. The rest of paragraph 5.6 shows how the d. g. f. can be expanded to O. F. M. 's. A special case is shown to reduce problems arising from the ordering of input letters.

**EXAMPLE 14.**

Suppose $\{x^0, x^1, \ldots \}$ and $\{y^0, y^1, \ldots \}$ are sets of variables from Systems of Type 2. Define the product of $ax^iy^j$ with $bx^my^n$ by

$$ax^iy^j \cdot bx^my^n = ab(x^i \cdot x^m)(y^j \cdot y^n).$$

Suppose $F = (S, \Lambda)$ is a T. F. M. Define $F$'s $(p, \Lambda, \Lambda^{-1})$ d. g. f. to be

$$\sum_{i=0}^{p} \sum_{j=0}^{p} a_{ij} x^i y^j$$

where

$$a_{ij} = \#((\Lambda^p(S) - \Lambda^p(i-1)(S)) \cap (\Lambda^{-i}(S^c) - \Lambda^{-i-1}(S^c))).$$

Consider the T. F. M.'s of Figure 11. The $(2, \Lambda_1, \Lambda_1^{-1})$ d. g. f. of $F_1$ is

$$a_1(x, y) = x^0y^0 + x^1y^1 + x^2y^2.$$
and the \((2, \Lambda_2, \Lambda_2^{-1})\) d.g.f. of \(F_2\) is

\[ a_2(x, y) = 2x^0y^0 + x^2y^1. \]

The \((2, [\Lambda_1, \Lambda_2], [\Lambda_1, \Lambda_2]^{-1})\) d.g.f. of \(F_1 \otimes F_2\) is

\[ a(x, y) = 2x^0y^0 + 2x^1y^1 + 2x^2y^1 + 3x^2y^2 = a_1(x, y) + a_2(x, y). \]

It is not difficult to see that the result \(a(x, y) = a_1(x, y) \cdot a_2(x, y)\) would have held independent of the particular \(F_1\) and \(F_2\) chosen.

**DEFINITION 23.**

Suppose \(\alpha\) is an \(m\)-tuple of non-negative integers, and \(\beta\) is an \(n\)-tuple of non-negative integers. Define

\[ \max (\alpha, \beta) \]

to be the \(mn\)-tuple whose \(n(i-1) + j\)th element is

\[ \max (a_i, b_j), \]

where \(a_i\) is \(\alpha\)'s \(i\)th element and \(b_j\) is \(\beta\)'s \(j\)th element.

For all \(n \geq 1 \leq n < \infty\), Definition 21(2) can be extended to the case where the exponents of the variables are \(n\)-tuples.

**DEFINITION 24.**

Suppose \(M = (S, \{0, 1, \ldots, m-1\}, \Lambda)\) is an O.F.M., \(\{x^\alpha | \alpha \text{ is an } n\text{-tuple of non-negative integers}, \text{ and } 1 \leq n < \infty\}\) is a set of variables of Type 2 (extended definition).

Define \(M\)'s \((p, \Lambda)\) d.g.f. \(a(x)\) by

\[ a(x) = \sum_{i=0}^{p} \sum_{j=0}^{p} \ldots \sum_{k=0}^{p} a_{ij\ldots k} x^i y^j \ldots k \]

where

\[ a_{ij\ldots k} = \#(\hat{\Lambda}_0^{p-i}(S) - \hat{\Lambda}_0^{p-(i-1)}(S)) \cap (\hat{\Lambda}_1^{p-j}(S) - \hat{\Lambda}_1^{p-(j-1)}(S)) \]

\[ \cap \ldots \cap (\hat{\Lambda}_m^{-k}(S) - \hat{\Lambda}_m^{-k-1}(S)) \]

\[ \ast \hat{\Lambda}_q \text{ has the meaning of Definition 16.} \]
Define M's \((p, \Lambda^{-1})\) d. g. f. \(A(x)\) by

\[
A(x) = \sum_{i=0}^{P} \sum_{j=0}^{P} \ldots \sum_{k=0}^{P} A_{ij\ldots k} x^{ij\ldots k}
\]

where

\[
A_{ij\ldots k} = \#(\widehat{\Lambda}^{-1}_0 (S^c_0) - \widehat{\Lambda}^{-1}_0 (S^c_0)) \cap (\widehat{\Lambda}^{-1}_1 (S^c_1) - \widehat{\Lambda}^{-1}_1 (S^c_1))
\]

\[
\cap \ldots \cap (\widehat{\Lambda}^{-k}_{m-1} (S^c_{m-1}) - \widehat{\Lambda}^{-k}_{m-1} (S^c_{m-1})).
\]

Let \(F = (S, \{0, 1, \ldots, m-1\}, \Lambda)\) and \(G = (T, \{0, 1, \ldots, n-1\}, \Gamma)\) be O.F.M.'s, and let \(P\) be the length of the longest maximal chain in any S.F.M. included in either \(F\) or \(G\).

Let \(H = (S \otimes T, \{0, 1, \ldots, mn-1\}, \Pi)\) be an O.F.M. where \(\Pi\) is defined by

\[
\Pi((s,t), q) = (\Lambda (s,i), \Gamma (t,j))
\]

where \(i\) and \(j\) are the unique numbers s.t. \(i \in \{0, 1, \ldots, m-1\}, j \in \{0, 1, \ldots, n-1\}\) and \(ni + j = q\). It is not too difficult to see that \(H\)'s \((p, \Pi)\) d. g. f. is the product of \(F\)'s \((p, \Lambda)\) d. g. f. and \(G\)'s \((p, \Gamma)\) d. g. f., and that \(H\)'s \((p, \Pi^{-1})\) d. g. f. is the product of \(F\)'s \((p, \Lambda^{-1})\) d. g. f. and \(G\)'s \((p, \Gamma^{-1})\) d. g. f.

6. CONCLUSION

The results for the decomposition of T.F.M.'s as the products of smaller T.F.M.'s suggest a decomposition scheme essentially due to Yoeli, for O.F.M.'s. If \(M = (S, X, \Lambda)\) is an O.F.M., then in the terminology of Definition 16,

\[
M = \sum_{x \in X} M_{x}^\pi.
\]

Let

\[
\eta = \{N \mid N \text{ is a S.F.M. and } E(x, N') \text{ and } (x \in X, \text{ } N' \text{ is a S.F.M. and } N \otimes N' \equiv M_x)\}.
\]

\(\text{*}_q^c\) is the set of cyclic states in \(M_q\)
and let the S. F. M.'s $N_i$ and $P_j$ be defined by

$$N_i = (T, i, \Gamma_i), \quad 1 \leq i < \infty$$

and

$$P_j = (W, j, \Pi_j), \quad 1 \leq j < \infty$$

If $M$ can be decomposed, then the decomposition must take the form

$$M \cong \sum_{i=1}^{m} N_i \otimes \sum_{j=1}^{n} P_j,$$

where:

1. Each $N_i$ or $P_j$ is isomorphic to some member of $\eta$.
2. $\#(X) = mn$
3. $\#(S) = \#(T) \cdot \#(W)$.

Given the set $\eta$, parts (2) and (3) above suggest shortcuts for decomposing $M$ so that not all possible combinations of factors from $\eta$ will be tried.

The other solution to the problem of decomposition of the O. F. M. is that of Hartmanis, with additional remarks in Hartmanis. The general method outlined by Hartmanis for decomposing an O. F. M. $M$ is to first find all of $M$'s partitions with substitution property. This can be done by identifying pairs of states or inputs, and determining what other identifications are necessary to have a partition with substitution property on $M$. The set of partitions with substitution property thus generated is a primitive set $Q$ from which all other partitions with substitution property on $M$ may be generated by taking sums (see Hartmanis) of partitions in $Q$. Once the set of all partitions with substitution property on $M$ has been obtained, pairs of these partitions satisfying conditions equivalent to those of Theorem 1 must be checked to see if they correspond to a decomposition of $M$.

It is difficult for the author to assess the relative merits of these two decomposition schemes. Both schemes could be easily programmed for a computer, and this is perhaps the best way to test the two procedures. A few comments perhaps are in order though.

Both schemes start out with the generation of sets of O. F. M.'s. In the Yoeli method, the set of S. F. M.'s $\eta$ is generated. In the Hartmanis method all, or practically all, of the homomorphic images of the O. F. M. $M$ are generated. Call this set $R$. 
Given the sets \( \eta \) and \( R \), it is easier to find decompositions of \( M \) from \( R \) than from \( \eta \), unless \( R \) is very much bigger than \( \eta \). This is because products of sums of S. F. M.'s from \( \eta \) must be tested for isomorphism to \( M \), whereas only products of O. F. M.'s from \( R \) need be tested for isomorphism to \( M \).

In the light of the comments above, it is obvious that for any benefits to accrue from the use of the Yoeli decomposition scheme over the Hartmanis decomposition scheme, \( \eta \) must either be much smaller than \( R \), or \( \eta \) must be much easier to generate than \( R \). It is to this latter problem that the author has addressed himself. The author feels that the methods suggested in this report for generating \( \eta \) are satisfactory enough to warrant further consideration of the Yoeli scheme.

Whether or not the Yoeli scheme is worthwhile, the results obtained in this report give some insight into the structure of F. S. M.'s, and particularly O. F. M.'s and S. F. M.'s. As an example, consider Corollary 1(2) to Theorem 9. This corollary shows that one thing, which can never be achieved by taking the product of two T. F. M.'s, is a T. F. M. whose longest maximal chain is longer than the longest maximal chain of both the original T. F. M.'s. This is an important limitation of product formation.

The material on generating functions shows that if given certain gross characteristics of two T. F. M.'s, some of the gross characteristics of their product may be determined quite easily. Conversely, if given a gross description of a T. F. M., some of the gross characteristics of the T. F. M.'s which may be factors of that T. F. M. can be determined quite easily.
Acknowledgments

The author has received help and encouragement from many sources. He wishes to acknowledge some of these sources.

The Applied Mathematics section of the Data Sciences Laboratories of the Air Force Cambridge Research Laboratories at Bedford, Massachusetts is to be thanked for allowing the author to work on his thesis. In particular the author wishes to thank Mr. William Lawlor of the Air Force Cambridge Research Laboratory for doing the illustrations in the original manuscript.

The typing of the original manuscript was done by Mrs. Claire Griffiths and Mrs. Rose Gifford. The author wishes to thank them for a job well done.

Finally the author wishes to thank his thesis advisor, Dr. C.L. Liu, for his help and suggestions. It was a pleasure working with Dr. Liu.
References

5. RHODES, J. L., An Algebraic Theory of Machines, informal notes on research done by the author in 1961.

Bibliography
