Vol. XV, 1 REVISTA MEXICANA DE FISICA 1966

LOGICAL ANALYSIS OF ORGANIZATION IN FINITE AUTOMATA

A. Medina
Comisión Nacional de Energía Nuclear

(Recibido: Junio 3, 1965)

RESUMEN

Partiendo de las suposiciones usuales relativas a la operación de autómatas finitos, se hace un análisis con medios formales puramente lógicos, de sus consecuencias. Los estímulos externos se asocian a un conjunto de relaciones entre los estados y con el auxilio del cálculo relacional se muestra que, relativamente a un alfabeto de estímulos, los estados pueden agruparse en sólo dos tipos de unidades de más alto orden. El mismo formalismo se usa para un análisis preliminar de los automorfismos y endomorfismos de autómatas. Se introduce el concepto de conmutador lógico y se hace ver que toda transformación endomórfica, tratada como una relación, debe satisfacer ciertas propiedades de conmutación que la caracterizan como una constante de movimiento.

conceptos fundamentales, así como para facilitar la lectura al investigador no familiarizado con la lógica formal, se ha introducido, en las secciones 2 a 6, un corto sumario de algunas definiciones y algunos teoremas bien conocidos de los cálculos predical y relacional, que pueden encontrarse en cualquier buen texto de lógica. De la sección 7 en adelante, se muestra la aplicación de éstas disciplinas al análisis de autómatas.

Partiendo de la noción de autómata finito definido por medio de sus ecuaciones canónicas, se hace notar que éstas pueden describirse mediante un conmunto de relaciones. La aplicación sistemática del cálculo relacional a este conjunto, permite obtener por medios puramente lógicos y muy simples, un cuadro general de la organización posible de los estados de un autómata, en unidades de orden más elevado, demostrándose, que son posibles solamente dos: el grupo y la familia. Cada una se puede definir mediante una relación adecuada, cuyas propiedades reflejan el caracter mismo del grado de organización descrito.

El mismo formalismo se emplea en las últimas secciones para efectuar un exámen preliminar de las propiedades de transformación del autómata que dejan invariante el sistema de relaciones inducidas por los estímulos externos. Tales son los endomorfismos y los automorfismos. Se introduce la noción de conmutador lógico y se demuestra que toda transformación que conmuta con todas las transformaciones inducidas por los estímulos, es, en general, un endomorfismo que puede caracterizarse como una constante de movimiento.

ABSTRACT

Starting from the usual assumptions concerning the operation of finite automata, an analysis of their consequences is carried out by formal, purely logical means External stimuli are associated with a set of relations among the states, and with the aid of the relational calculus it is shown that, relative to an alphabet of stimuthe states may be grouped in only two kinds of higher-order units. The same formalism is used for a preliminary analysis of the automorphisms and endomorphisms of automata. The concept of logical commutator is introduced; it is shown that every

endomorphic transformation, considered as a relation, must satisfy certain commutation properties which characterize it as a constant of motion.

1. Introduction

This is the first of a series of papers devoted to the study of the theory of finite automata under a purely logical point of view.

Just as the well-known method of propositional calculus proved to be a valuable tool for analysis, design and general understanding of digital machines, one finds that a logical treatment incorporating other branches of logic such as predicate and relational calculus seems to be a powerful weapon, not only for reformulating known facts in a simple systematic manner, but also for penetrating into the deepest properties of automata which are not so apparent nor easily analyzed, when handled by ordinary algebraic methods.

The least advantage that can be argued in favour of the logical methods to be considered is their extreme simplicity. This is probably due to the constant use of a formal language in which the process of deduction and inference appears explicitly exhibited in a precise manner, thereby leading one, so to say, by the hand, to the consequences. The formalism contains in itself the machinery required for demonstrations and obtention of conclusions. Once the elements of the analysis are formally written, in many cases the conclusions practically spring off from the formalism by themselves.

In dealing with stimulus-induced transitions in automata and with certain of their transformation properties, relational calculus was used throughout. This not only allows a clear understanding of the structure of automata, and the nature of many of their not so obvious properties but, as will be shown in the third and fourth paper of the series, leads naturally to simple design and synthesis procedures.

The power of relational calculus for dealing with automata problems probably depends on a very basic characteristic: Structure and behaviour of automata are mostly dependent on transformations of sets of states. These transformations generally have no inverse in the ordinary algebraic sense, most of them cannot be described by group operations, nor by anything similar to the familiar type of linear algebras. This, it is believed, prevents in this type of work the use of mathematical systems which in other branches of applied mathematics have shown to possess beautiful possibilities for penetrating very deeply into the fundamental nature of things.

The inability of group-theoretical methods to deal with general automata problems is offset to a great extent by relational calculus. Here one naturally deals with every type of transformation, one-or many-valued, uniform or multiform. Nevertheless, where transformations cannot be algebraically inverted, they can be logically inverted. In fact, we shall see that one can use formal logical inverses in a way very similar to ordinary algebraic inverses. This is perhaps, among others one of the main characteristics that make relations so useful in the formal treatment of these problems.

Since this work will be shown in future papers to have some practical applications, the author considered that it would be convenient to make the whole treatment accessible to the practical engineer working in the field of cybernetics. To the author's knowledge, one does not often find engineers having a previous working experience of relational calculus or lattice algebra. On the other hand the whole treatment becomes so simple, once the general foundations of this mathematical system are grasped, that it appeared convenient to introduce this and future papers by a short summary ot predicate and relational calculus. This is the purpose of sections 2 to 6 of this work.

These sections are included only for the benefit of readers not acquainted with the formalism used, to fix, once and for all future reports, the notational conventions that are used in a field in which unfortunately there is no universally accepted symbolism. Besides, it was the author's purpose to bring out the important facts, saving the unacquainted reader the trouble of digging them up from the huge existent literature on the subject. It is hoped that the learned scholar who will not find anything new in this brief summary, will patiently endure this, for him boring, exposition of well-known facts on behalf of the less conspicious reader. Since the material is contained in current text books on the subject, no proofs are offered, but reference to standard texts are given, without explicit and overloading mention of individual papers.

2. Classes and properties[1,2]

Suppose we have a class or collection 1 of things and let x be a variable on that class, that is, x designates an indeterminate member of the class. We write $1 = \{x\}$, meaning that 1 is the set of elements that x can designate. In our discussion, whenever we talk about subsets, it will be understood that we refer to subsets of our fundamental "universal class" 1.

Let $f, g, h, \ldots\ldots$ denote properties of the objects of 1. Each such property determines a subset of 1, namely, the subset of objects of 1 which possess the corresponding property. The symbol "$f(x)$" means that the indeterminate object x possesses the property f, and will be read "x is f". When specialized to individual objects, the formula "$f(x)$" becomes a proposition that may be true or false according to whether the selected particular object does or does not possess the property f.

f by itself is the name of the property, and as far as the universal class remains fixed, can be understood as a concept of the class of objects possessing the property. Thus, we identify f with the abstract

$$f = \hat{x}\, f(x) \tag{1}$$

that is, the class of objects x which are f. In our discussion "$f(x)$" and "$x \in f$" shall have the same meaning, and every formula or proposition of the predicate calculus, can be translated into a corresponding expression of the algebra of classes.

A list of symbols of the predicate calculus, their meaning as well as their equivalence in the algebra of classes is offered.

Predicate Calculus	Algebra of Classes	Meaning
$f(x)$	$x \in f$	x is f
$\sim f(x)$ or $f'(x)$	$x \in f'$	x is not f
$(\exists x)\, f(x)$	$f \neq \Lambda$	there is an f, there are f's
$\sim(\exists x)\, f(x)$	$f = \Lambda$	there is no f
$(x)\, f(x)$	$f = 1$	all are f
$\sim(x)\, f(x)$	$f \neq 1$	not all are f
$f(x) \cdot g(x)$	$x \in f \cup g$	x is f and g
$f(x) + g(x)$	$x \in f \cup g$	x is f or g
$(x)\,[f(x) \supset g(x)]$	$f \subseteq g$	all f's are g's
$(x)\,[f(x) \equiv g(x)]$	$f = g$	the f's are precisely the g's
$\hat{x}\,[f(x) \wedge g(x)]$	$f \wedge g$	the f's that are not g's and the g's that are not f's

Here as usual " . " denotes conjunction, " + " alternation, " $\supset$ " the conditional, " $\equiv$ " the biconditional, " $\wedge$ " the disjunction or exclusive alternation.

For a finite class 1, having elements $x_1, x_2, \ldots\ldots x_n$, the existentia and universal quantifiers mean

$$(\exists x)\, f(x) = f(x_1) + f(x_2) + \ldots + f(x_n)$$

$$(x)\, f(x) = f(x_1) \cdot f(x_2) \cdot \ldots \cdot f(x_n) \tag{2}$$

3. Relations[1,2]

From any two elements of our universal class, x and y, we can form an ordered pair $x;y.$, namely, a set containing the objects x and y only, in which an ordering has been introduced, whereby one considers x to be the first and y the second element of the set. The collection $\{x;y\}$ of all ordered pairs of element- of 1 is the so-called direct or topological second power of 1 and denoted by $1 \times 1 = 1^2$. A relation r is a property on 1^2, namely a subset of ordered pairs of elements of 1. The ordered pair $x;y$ belongs to r if and only if x bears the relation r to y, which we write $r(x,y)$ and read for simplicity "x is a r of y". In symbols

$$
\begin{aligned}
r(x,y) &= (x;y) \in r \\
r &= \hat{x}\,\hat{y}\,r(x,y)
\end{aligned}
\tag{3}
$$

x is the ***referent*** and y the ***relatum***.

Because of the fact that relations are subsets of a certain class, they possess all ordinary properties of classes, that is, they form a Boolean algebra under the relations of equality and inclusion, and the operations of complementation, intersection and union. Thus two relations are equal if and only if they contain the same ordered pairs.

If r and s are relations, $r = s$ means

$$(x)(y)\,\big(r(x,y) \equiv s(x,y)\big) \tag{4}$$

$r \subset s$ means

$$(x)(y)\,\big(r(x,y) \supset s(x,y)\big) \tag{5}$$

r', the complement of r, is the set of ordered pairs not contained in r, that is,

$$r' = \hat{x}\,\hat{y} \sim r(x,y) \tag{6}$$

The union $r \cup s$ of two relations, is a set of pairs in which the first member of the pair is a r, or a s, or both, of the second:

$$r \cup s = \hat{x}\,\hat{y}\,\big(r(x,y) + s(x,y)\big) \tag{7}$$

The intersection of r and s, $r \cap s$, is the set of ordered pairs in which the first member bears the two relations to the second:

$$r \cap s = \hat{x}\,\hat{y}\,\big(r(x,y) \,\dot{\cdot}\, s(x,y)\big) \tag{8}$$

Under these operations, relations obey the theorems of Boolean algebra. In particular one defines a null relation $\dot{\Lambda}$ denoting the empty class of ordered pairs and a universal relation $\dot{1} = 1^2$ being the collection of all ordered pairs. These two particular relations are the universal bounds of the Boolean algebra of relations. Every relation r fulfills

$$\dot{\Lambda} \subset r \subset \dot{1} \tag{9}$$

Due to the circumstance that the elements of a relation are pairs having an ordering, it is possible to define operations on relations which depend precisely on this ordering.

We define the inverse $\breve{r}$ of a relation r by

$$\breve{r}(x,y) = r(y,x) \tag{10}$$

that is, the ordered pair $x;y$ belongs to $\breve{r}$ if and only if the inverse pair $y;x$ belongs to r.

$\breve{r}(x,y)$ will be read "x is a r inverse of y" or, perhaps better, "y has x for a r".

Further, the "relational product" rs of two relations r and s is defined through

$$rs\,(x,y) = (\exists z)\,\big(r\,(x,z)\;.\;s\,(z,y)\big) \tag{11}$$

that is, "x is a r of a s of y".

In abstract form

$$rs = \hat{x}\,\hat{y}\,(\exists z)\,\big(r\,(x,z)\;.\;s\,(z,y)\big) \tag{11a}$$

Relational inversion is obviously involutory:

$$\breve{\breve{r}} = r \tag{12}$$

The relational product is associative

$$r\,(st) = (rs)\,t = rst \tag{13}$$

but in general not commutative

$$rs \neq sr \tag{14}$$

The relational power r^n of a relation r can be recursively defined through

$$\begin{aligned} r^1 &= r \\ r^{k+1} &= r^k r = r r^k \end{aligned} \tag{15}$$

We shall denote by I the identical relation, that is, the relation that any object bears to itself

$$I = \hat{x}\,\hat{y}\,(x = y) \tag{16}$$

I is merely the collection of ordered pairs $x;x$ and the formula $I(x,y)$ has the same meaning as $x = y$. We shall extend our definition of powers to zero exponent by adopting the convention

$$r^0 = I \tag{15a}$$

A relation s is said to be symmetric if

$$s = \breve{s} \tag{17}$$

that is if

$$(x)(y)\,\big(s\,(x,y) \equiv s\,(y,x)\big) \tag{17a}$$

A relation r is said to be reflexive if

$$I \subset r \tag{18}$$

which means

$$(x)(y)\,\big(x = y\,.\supset r\,(x,y)\big) = (x)\,r\,(x,x) \tag{18a}$$

A relation t is transitive if

$$t^2 \subset t \tag{19}$$

that is if

$$(x)(y)(z)\big(t(x,z)\,.\,t(z,y)\,.\supset t(x,y)\big)$$

(19a)

An operation which will be important in our work, is the so called "ancestral", defined by

$$\#r = I \cup r \cup r^2 \cup r^3 \cup \ldots\ldots = \bigcup_{k=0}^{\infty} r^k \qquad (20)$$

$\# r(x,y)$, "x is a r ancestral of y", means that x is identical to y, or x is a r of y, or x is a r of a r of y, or ...

The operation

$$\natural r = r\#r = r \cup r^2 \cup r^3 \cup \ldots\ldots = \bigcup_{k=1}^{\infty} r^k \qquad (21)$$

is called the "proper ancestral".

4. Graphical Representation of Relations

In a finite universal class 1, each element can be represented by a dot or, still better, a small circle, in which the name of the element is inscribed. A property f can be pictured in this representation by drawing a closed curve enclosing all elements possessing this property and writing somewhere along the curve the name of the property (or a symbol for it). In such a diagram a relation r can be represented as follows:

Whenever $r(x,y)$ is true, one draws an arrow, named r, from the circle representing the element y to the circle representing the element x (Fig. 1).

Representation of r(x,y) Representation of r(x,x)

Fig. 1

If x bears the relation r to itself one draws a loop going from x to x.

The set of arrows named r is a graphical representation of the relation r. The purpose of labelling arrows with the name of the relation is that of allowing the simultaneous representation of several relations on the same diagramme. If only one relation were involved, such labelling could of course be omitted.

The representation of the inverse relation $\breve{r}$ is the same as that of r, except that all arrows are inverted. For this reason r and $\breve{r}$ are not separately represented.

If $r(x,y)$ and $r(y,x)$ are true, two arrows are used (Fig. 2).

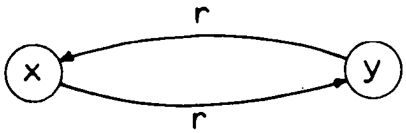


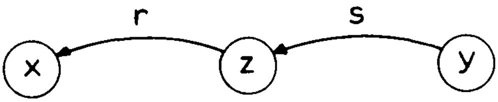
Fig. 2

If x bears to y relations r and s, that is, if x is a $r \cap s$ of y, one may use separate arrows for r and s, or use a single arrow with both names on it (Fig. 3).

Fig. 3

The graphical representation of $r \cup s$ is the set of all arrows bearing r or s or both.

The relational product appears automatically represented in our diagramme because $rs(x,y)$ will be true whenever one can find a z having x for an r, and being an s of y, that is, whenever one can find at least one path from y to x consisting of two pieces, the first named s and the second r (Fig. 4).



rs(x,y)

Fig.4

Two equal relations would be represented by exactly the same set of arrows If $r \subset s$, then every arrow named r will be paralleled by an arrow named s.

5. Projections

If f is any class of ⅂ and r a relation on Ͱ, the formula $r(x,y) . f(y)$, "x is a r of y and y is f", gives rise to a property of x:

$$(\exists y)(r(x,y) . f(y))$$

namely, that of being an r of an f.[†]

The set

$$r"f = \hat{x}(\exists y)(r(x,y) . f(y)) \qquad (22)$$

[†]Such a property is very common in everyday language in expressions such as "wife of a soldier", "father of a pupil", "friend of a politician", etc.

of all x which are r's of at least one f, namely "the r's of the f's" is called the *projection* of class f by relation r.

In a relational diagramme such a class consists of all circles which are terminal points of r-arrows, whose starting points are in f (Fig. 5).

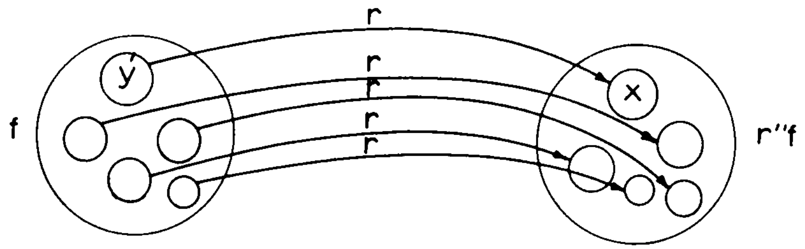


Fig.5

The projection of a class f by the inverse relation $\breve{r}$, namely

$$\breve{r}'' f = \hat{x}\ (\exists y)\ (f(y)\ .\ r(y,x)) \qquad (23)$$

the set of all x having an f for an r, is called the *retrojection* of class f by relation r. Graphically $\breve{r}''f$ is the set of all circles which are starting points of r-arrows terminating in f. Since $r''f$ and $\breve{r}''f$ determine certain classes in ⅄, both can be used as predicates in the formulas.

$$\begin{aligned} r''f(x) &= x \in r''f \\ \breve{r}''f(x) &= x \in \breve{r}''f \end{aligned} \qquad (24)$$

If one uses as f the universal class ⅄, one obtains the classes:

$$r''\mathbf{1} = r'' = \hat{x}\,(\exists y)\, r(x,y)$$

$$\breve{r}''\mathbf{1} = \breve{r}'' = \hat{x}\,(\exists y)\, r(y,x) \qquad \ldots\ldots \qquad (25)$$

r'' is called the *direct domain* of r and represents the set of all elements which are r of some element. $\breve{r}''$, the *inverse domain* of r, is the set of all x having an r.†

Again, r'' and $\breve{r}''$ can be used as predicates in the formulas $r''(x)$, x is an r, and $\breve{r}''(x)$, x has an r. Another important type of projection is obtained when f is a unitary class. By ***unitary class*** we understand a class consisting of just one element. Thus, the unitary class of x is the class containing the sole member x, and is denoted by ιx. It is convenient to distinguish between the element x and the class ιx. For example, if x has a property f, we write $x \in f$, but $\iota x \subset f$. In the first case we talk about an element of f, in the second we refer to a subset of f. Every class f is the union of all unitary classes of its elements.

The projection and retrojection

$$r''\iota y = \hat{x}\,.\, r(x,y)$$

$$\breve{r}''\iota y = \hat{x}\,.\, r(y,x) \qquad \ldots\ldots \qquad (26)$$

describe the r's of y and the elements having y as an r respectively. Once more, both expressions can be used as predicates, but in this case such use would be trivial, because $r''\iota y(x)$ and $\breve{r}''\iota y(x)$ mean exactly the same as $r(x,y)$ and $\breve{r}(x,y)$.

A very important particular case arises when there is a unique x fulfilling the condition. For ex., the unique x being an r of y is called *the* r of y and denoted by: $r'y$. If y is r of a unique element, this element is described by $\breve{r}'y$.

---

†In conversational language such classes are widely used as "the wives", "the fathers", "the friends", etc.

This particular type of projection is called a relational description.

$$r' y = (\iota x)\, r(x, y)$$

$$\breve{r}' y = (\iota x)\, r(y, x) \qquad \ldots\ldots\ldots \quad (27)$$

(ιx) is the *description operator* meaning "the only x such that".

In case of uniqueness, one still must distinguish between $r' y$ and $r'' \iota y$. One is related to the other by:

$$r' y = (\iota x)\left(r'' \iota y(x)\right)$$

$$r'' \iota y = \iota(r' y) \qquad \ldots\ldots \quad (28)$$

and the same for $\breve{r}$.

A class f is said to be closed under a relation r if

$$r'' f \subset f \qquad (29)$$

that is, if all the r's of the f's are f's. If f is closed under r one says that property f is *hereditary* under r, or that r "transmits" property f.

It can be easily shown that if a class is closed under r, it will be closed under any power of r.

$$r^{n}{}'' f \subset f \qquad (29a)$$

and under both ancestrals

$$\# \, r'' f \subset f$$

$$\natural \, r'' f \subset f \tag{30}$$

that is, if the r's of the f's are f's, then all r-descendants (or ancestors) of the f's will, in turn, be f's.

6. Identity Relations and Equivalence Classes.

Any reflexive, symmetric and transitive relation in mathematics is called an *identity relation*, because ot the fact that it can be used to define some kind of equivalence between the elements of a set and because it is closely connected with the fundamental idea of a group of transformations.

In our work we shall find a wide use for such a type of relation, in particular with the formation of the so-called "equivalence classes".

Let r be a relation over our universal set $\mathbf{1}$, which we assume to be:

a) reflexive $I \subset r$, b) symmetric: $r = \breve{r}$, c) transitive: $r^2 \subset r$

Let $x_1 \in \mathbf{1}$ be a specified element of $\mathbf{1}$.

The set

$$R(x_1) = r'' \lfloor x_1 = \breve{r}'' \lfloor x_1 \tag{31}$$

is the set of all elements of $\mathbf{1}$ which bear to x_1, relation r. Because of the transitive character of r any two elements of this set bear to each other the same relation. Thus, if x_1', $x_1'' \in R(x_1)$, then $r(x_1', x_1)$ and $r(x_1, x_1'')$, are both true.

But transitivity requires that

$$r(x_1', x_1) \; . \; r(x_1, x_1'') \; . \supset r(x_1', x_1'')$$

as stated.

Evidently, $R(x_1) \subset V$ is a subset of V. If $R(x_1) = V$, then the whole set V consists of a single class of elements equivalent under r. Suppose, however, that $R(x_1) \neq V$, then there exists an element x_2 such that $x_2 \in V$ but $x_2 \notin R(x_1)$. Again, x_2 cannot bear relation r to any element of $R(x_1)$. Let now

$$R(x_2) = r'' \iota x_2 = \breve{r}'' \iota x_2 \qquad (31a)$$

be the set of elements related to x_2 by r. Clearly $R(x_1) \cap R(x_2) = \Lambda$. The two classes thus formed have no element in common.

Now, $R(x_1) \cup R(x_2) \subset V$. The process can be continued and one finds a numerable set of r-inequivalent elements $x_1, x_2, x_3, \ldots$ and a numerable set of r-classes $R(x_1)$, $R(x_2)$, $R(x_3), \ldots$, no two classes of the set having common elements.

If our universal class V is finite, one can find at most a finite number of such classes whose union will eventually exhaust the whole set V. In this manner V will be decomposed in a finite number of r-classes

$$V = R_1 \cup R_2 \cup \cdots \cup R_n \qquad (32)$$

any two being disjoint

$$R_i \cap R_j = \Lambda, \text{ if } i \neq j \qquad (32a)$$

$$i, j = 1, 2, \ldots n$$

Such a decomposition is *exhaustive*, namely, any element of V belongs to a certain class, and *exclusive*, that is, an element belongs to one and only one class.

It is also clear, that the same decomposition is obtained irrespective of the particular r-inequivalent elements $x_1, x_2, \ldots x_n$, originally used for its construction. Any set of elements x_i $(i = 1,2, \ldots n)$ such as $x_i \in R_i$ will lead to the same decomposition. The r-classes R_i are completely determined by r itself and depend only on the nature of the relation, with the proviso that it be reflexive, symmetric and transitive. A decomposition of the sort just described, is called a *partition*. Because of its origin, we shall denote it as a r-partition. The classes R_i are the members of the partition.

As an example of this type of process, consider a property f defined on $\mathfrak{T}$. We can define a relation on $\mathfrak{T}$ with the aid of this property, by considering that two elements possessing the property f are f-related. This particular relation we denote by $\dot{f}$, and define through

$$\dot{f}(x,y) = f(x) \cdot f(y) \quad \ldots\ldots \tag{33}$$

$\dot{f}(x,y)$ can be read: "both x and y are f". Similarly, from the negation of the property one defines $\dot{f}'$. $\dot{f}'(x,y)$ means that neither x nor y are f. Now the relation $\dot{f} \cup \dot{f}'$ is reflexive, symmetric and transitive, and can be used to effect a decomposition of $\mathfrak{T}$ of the type just discussed. In this case, however, the generated partition is particularly simple. It consists of two members: class f itself, containing all elements possessing the property, and its complement f' containing all elements that do not possess the property.

The decomposition

$$\mathfrak{T} = f \cup f' \tag{33a}$$

effected by

$$\dot{f} \cup \dot{f}' \tag{33b}$$

is the simplest type of non-trivial partition, namely a *dichotomy*. It is permissible to assume that any dichotomic identity relation is a concept of a property.

Let now r be an arbitrary relation on Υ. Out of this r, one can always form two relations, $r\breve{r}$ and $\breve{r}r$, illustrated in Fig. 6.

Fig.6

Two elements x and y bearing relation $r\breve{r}$ to each other are such that both are referents of the same element; that is, co-*referents*. If x and y are connected by $\breve{r}r$, both are relata of the same element, that is, *correlata*. The first relation is called co-reference, the second is called *correlation*. We shall find frequent use for both types of relations.

7. States and Relations in Automata

A finite automaton A (Fig. 7)

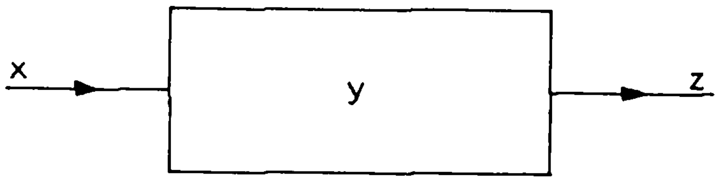


Fig.7

will be pictured according to the usual conventions: We imagine a finite set of input lines that at any time may present any of a finite set of states. Each such state is an input symbol, the set itself is an input alphabet $\mathfrak{X}$ and x is a variable on $\mathfrak{X}$. The automaton possesses a finite number of output lines, which in turn are capable of presenting any of a finite number of possible states, each such state being an output symbol, the totality of symbols being the output alphabet of A, $\mathfrak{Z}$, and z a variable over this alphabet. Finally A can be found in any of a certain finite number of internal states, the general state being represented by y, whose totality is the so-called *phase-space* $\mathfrak{Y}$ of A.

We further assume that an adequate unit of time has been adopted, in such a manner that during the interval of one unit of time only one state or symbol can be present and any possible changes can occur only from one unit of time to the next. No changes of states will occur within a unit of time. In this manner our time variable t will assume discrete integral values. We want to think of these values as names for the intervals themselves, selected after a first interval has been arbitrarily specified. We shall not interpret the values of t as hypothetical sampling instants.

The behaviour of the automaton will be characterized by the so-called *canonical equations*[3], which will be assumed of the form

$$\begin{aligned} y_{t+1} &= f(y_t, x_t) \\ z_t &= g(y_t) \end{aligned} \qquad (1)$$

The first uniquely determines the next state at interval $t+1$, in terms of the inner state y_t and the stimulus x_t prevailing at interval t. This equation (or rather set of equations), as is well known, determines a sequence of transitions of A, caused by a sequence of input symbols, starting from some initial state. The second equation relates the output symbol or decision with the inner state. For reasons which will be apparent in our analysis, no immediate dependence of z

on the input symbol x need be assumed. It is clear that the overt behaviour of A, as represented by z, will depend on the stimuli x, but it will be found that it is enough to assume that this dependence occurs through that of the inner states of A.

In order to simplify our notation, the variable t, which we shall translate as "the moment t", understanding by moment our basic unit of duration, will be tacitly assumed. Thus equation (1) will be written in the form

$$\begin{aligned} y^1 &= f(y,x) \\ z &= g(y) \end{aligned} \qquad (1a)$$

The input symbols of $\mathfrak{X}$ will be labelled as $x_1, x_2, \ldots x_m$, those of $\mathfrak{Y}$ as $y_1, y_2, \ldots y_n$ and those of $\mathfrak{Z}$ as $z_1, z_2, \ldots z_p$. Let the reader be reminded that x, y, z are variables over the corresponding sets and designate a not specified symbol of their respective domain.

If at some moment A is in a state y, under the stimulus x, according to the canonical equations (1), it will undergo a definite transition to some state y^1. This transition can be represented as usual in a kinematic diagram[3] as shown in Fig. 8.

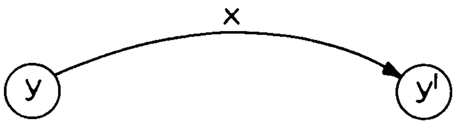


Fig.8

We can assume that the input symbol x defines a relation among the elements of the phase-space of A, whose graphical representation is provided by the kinematic diagram itself. If under symbol x, A goes from state y to state y^1, we say that y^1 is an x-sequent of y and bears to it the relation x. Consequently, we write $x(y^1, y)$. Fig. 8 is the graph of this relation.

As with every relation, we can define the inverse $\breve{x}$. If y^1 is an x-sequent of y, then y is an x-precedent of y^1, which we can write as $\breve{x}(y, y^1)$ †

The input alphabet $\mathfrak{X}$ can be interpreted as a set of sequency relations in phase-space of A. The complete kinematic diagram of A can be interpreted as the graphical representation of this set of relations according to the conventions previously established (pág. 11).

It is convenient for our purposes to explicitly state certain assumptions contained in the canonical equations, because we want to explore the consequences of the logical content of the equations themselves, without specific reference to their formal expression.

We assume A to be *determinate* , by which we understand that, given any state y and any stimulus x, we know the particular transition experienced by A under these circumstances. Determinateness in the sense we have just defined, can be conveniently expressed by either of two synthetic expressions, one in the formal notation of predicate calculus, the other in the notation of relational calculus;

$$(x)(y)(\exists y^1)\, x(y^1, y)$$

..... (2)

or

$$\breve{x}``\mathfrak{Y} \subset \mathfrak{Y}$$

A is also assumed to be *definite*, by which we understand that, for any x and for any y, the x-sequents of y are all in $\mathfrak{Y}$.

$$(x)(y)(\exists y^1)\left(x(y^1, y) \supset .\, y^1 \in \mathfrak{Y}\right)$$

...... (3)

$$x``\mathfrak{Y} \subset \mathfrak{Y}$$

---

†Note: In order to be rigorous it should be necessary to distinguish between "x" as a variable over the symbols of an alphabet and the corresponding induced relation in the phase-space of A. This we could do by using for this concept the notation "$\dot{x}$". However, as long as there is no danger of confusion, the same symbol "x" will be used, the concept being clarified in all cases by the context.

Finally, it will be assumed that A is *causal*, that is, that given an initial state y and an input symbol x, there is a unique y^1 to which A goes, when in state y is stimulated by x. According to the theory of relations, we can speak of *the* x-sequent of y, designated by the relational description $y^1 = x'y$. In other words the class $x'' \llcorner y$ of x-sequents of y is a unitary class containing a single element. Causality can be expressed by the formula

$$(x)(y)\left(x'' \llcorner y = \llcorner (x'y)\right)$$

$$\text{or} \qquad x\breve{x} \subseteq I \qquad (4)$$

According to the forgoing assumptions, in the kinematic diagram of A there will be m lines starting from each node y, one for each symbol of the input alphabet. Each line will go to one and only one y^1. In particular this y^1 can be the original state y itself, in which case, the line loops around y. When this is the case, one says that y is *stable* under x. Condition of stability is then that

$$x'y = y \qquad (5)$$

or that $x(y, y)$ be true.

A state y_j will be said to be a *sequent* of y_i if there exists a symbol x causing a transition from y_i to y_j. This relation of sequence we denote by s and define by

$$s(y_j, y_i) = (\exists x)\, x(y_j, y_i) \qquad (6)$$

Since on a finite set such as $\mathfrak{X}$ the existential operator $(\exists x)$ amounts to an alternation of the operand over all elements of the set, the forgoing expression merely defines relation s as the union of the m individual relations corresponding to the particular input symbols.

$$s = x_1 \cup x_2 \cup \cdots\cdots \cup x_m = \bigcup_{k=1}^{m} x_k \tag{7}$$

In short, a state y_j will be a sequent of y_i if there exists an arrow in the diagram, going from y_i to y_j, irrespective of the name of the arrow. Sequent, in this sense, could be translated as "possible sequent".

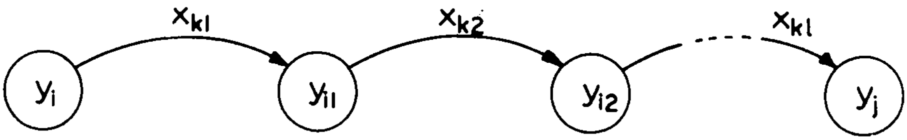
The inverse relation $\breve{s}$ is that of precedence and from (7) corresponds to

$$\breve{s} = \breve{x}_1 \cup \breve{x}_2 \cup \cdots\cdots \cup \breve{x}_m = \bigcup_{k=1}^{m} \breve{x}_k \tag{7a}$$

A sequence of l input symbols will be called a word of length l. This word can be interpreted as the name for an external definite event of duration l.

Within the automaton, this word will cause a set of l transitions from a given initial to a definite final state.

Just as each individual symbol can be represented by a relation or set of ordered pairs of states, whose first member is an initial and whose second member is the final state of the transition, each l-word can be equally well represented by a relation over the phase-space of the automaton, this relation being again the set of ordered pairs, whose first element is the initial state and whose second element is the state attained by A as a consequence of the word (Fig. 9)



word $\nu = x_{k1} x_{k2} \ldots x_{kl}$

Fig. 9

If the word consists of the symbols $x_{k_1}, x_{k_2}, \ldots x_{k_l}$ in this order, we write

$$w = x_{k_1} x_{k_2} \ldots x_{k_1}$$

The associated relation can be denoted by $w(y_j, y_i)$, meaning that y_j is a w-sequent of y_i. However, it is clear that in order to be y_j a w-sequent of y_i, one should have a chain of $l + 1$ states starting from y_i and ending in y_j connected by arrows which, in succession correspond to the letters of the word. This, however, means that one has to take a series of x-sequents as indicated by the word itself.

Starting with y_i one goes to the x_{k_1}-sequent, then to the x_{k_2}-sequent of this sequent, etc. Now then, this is precisely the definition of the relational product,

$$x_{k_l} x_{k_{l-1}} \ldots x_{k_2} x_{k_1} \ .$$

Word relation corresponds then to the relational product of the letters of the word in inverse order. We can define this relation by

$$w = x_{k_l} x_{k_{l-1}} \ldots x_{k_2} x_{k_1} \qquad (8)$$

According to the definition, the symbol of the word itself will be used to designate the relational product in inverse order.

A state y_j will be called a *sequent of order l* or shortly a *l-sequent*, of a state y_i, if there exists a l-word relating y_i to y_j, that is, if there exists in the kinematic diagram a path of length l going from y_i to y_j.

As in the definition of sequents, it is clear that the relation just derived, is the union of all the m^l relations corresponding to the m^l possible different words of length l with our m-symbol alphabet. This relation, however, has a very simple expression in our formalism: it is merely the lth relational power of s, namely,

$$s^l = \bigcup_{k_1, k_2, \ldots, k_l = 1}^{m} x_{k_1} x_{k_2} \ldots x_{k_l} \tag{9}$$

s^l is then the sequence of order l, or l-sequence relation. Similarly, $\breve{s}^l$ represents the relation of precedence of order l.

With the aid of s and its powers, we can define a more general type of relation among the states of automata, that of *succession*. A state y_j is said to be a *successor* of a state y_i, if there exists some word leading the automaton from y_i to y_j, that is, if for some $l (l = 1,2,3 \ldots)$ y_j is a l-sequent of y_i. This relation, then, will hold if y_j is a s or a s^2 or $s^3 \ldots$ of y_i, i.e., if y_j is a s-ancestral of y_i

$$\natural s = s \cup s^2 \cup s^3 \cup \ldots . = \bigcup_{l=1}^{\infty} s^l \tag{10}$$

For our purpose, we shall include in our succession relation the identity term $I = s^0$, by using the improper ancestral

$$\sigma = \# s = I \cup \natural s = \bigcup_{l=0}^{\infty} s^l \tag{11}$$

which means that we consider any state as its own successor, whether it has a looping branch or not. The inverse $\breve{\sigma} = \# \breve{s}$ represents *precession*. σ can be easily shown to be reflexive and transitive, that is, it fulfills

$$I \subseteq \sigma \qquad \sigma^2 \subseteq \sigma \tag{12}$$

It is clear that, in a finite automaton having n states, one does not have to consider sequence relations of orders greater than $n-1$, as far as the possibility of connecting one stade with another by a series of transitions induced by words, is concerned. In this sense one could limit the unions in equations (10) and (11) up to terms of order $n-1$. However, in considerations involving words as names of external events, we shall find convenient to retain the totality of terms as defined, in order to be able to translate events of indefinite duration into the *language of relations within the automaton.*

8. Connections and Groups

Two states of the automaton will be said to be *connected,* if each one of them is a successor of the other. The connection relation is then

$$c_0 = \natural s \cap \natural \breve{s} \tag{13}$$

and can be easily seen to be symmetric and transitive.

$$c_0 = \breve{c}_0 \qquad c_0^2 \subset c_0 \tag{14}$$

If y is any state of the automaton connected to at least another state, it will be true that:

$$(E\, y^1)\, c_0\, (y, y^1) = (E\, y^1)(\natural s(y, y^1)\ .\ \natural s(y^1, y)) = \natural s \natural s(y, y) \quad \ldots \quad (a)$$

But $\square s$ as an ancestral is a transitive relation, that is,

$$\natural s \natural s \subset \natural s \qquad \ldots \quad (b)$$

Then, $\natural s \natural s(y, y)\ .\supset .\ \natural s(y, y)$

and further

$$\natural s(y,y) = \natural \breve{s}(y,y) \qquad \text{(c)}$$

From a) b) and c) we conclude

$$(\exists y')\, c_0(y,y') \supset . \natural s(y,y) \,.\, \natural \breve{s}(y,y)$$

and, remembering the definition of c_0 given in (13),

$$(\exists y')\, c_0(y,y') \supset . \, c_0(y,y)$$

or in relational notation

$$c_0 \neq \dot{\Lambda} . \supset . \, I \subset c_0 \qquad \text{(15a)}$$

Whenever c_0 is non void, it is reflexive. By this reason, in defining connection relations in the automaton, we can use a connection defined through the improper ancestral as

$$c = \# s \cap \# \breve{s} = \sigma \cap \breve{\sigma} \qquad \text{(16)}$$

Remembering that $\# s = I \cup \natural s$ we can write

$$c = (I \cup \natural s) \cap (I \cup \natural \breve{s}) = I \cup (\natural s \cap \natural \breve{s})$$

or

$$c = I \cup c_0 \qquad \ldots\ldots \qquad \text{(17)}$$

This connection relation is

$$\begin{array}{lll} \text{reflexive} & I \subset c & \\ \text{symmetric} & c = \breve{c} & \quad (17a) \\ \text{transitive} & c^2 \subset c & \end{array}$$

and therefore, is an identity relation. As shown before, all relations of this kind, determine a partition of the elements of their domain into a set of equivalence classes. Thus, for any state y the set $c" \lfloor y$ is the set of all states connected to y. This set we call the *group of y*.

$$G(y) = c" \lfloor y \tag{18}$$

Every state $y^1 \in G(y)$ is connected to y, i.e. is both a successor and precessor of y. Two states $y_1, y_2 \in G(y)$ not only are connected to y, but, because of transitivity, are connected to each other. The set of states of $G(y)$ is a class of states having the property that, any two states of the set are successors of one another. It may happen that $G(y)$ contains a single state y.

We can associate to each state y of $\mathcal{Y}$ a group $G(y)$. If y_1 and y_2 are two different states, the relation between the associated groups $G(y_1)$ and $G(y_2)$ is evidently

$$G(y_1) = G(y_2) \;\; \text{if} \;\; c(y_1, y_2)$$

or

$$G(y_1) \cap G(y_2) = 0 \;\; \text{if} \;\; \sim c(y_1, y_2)$$

It is possible to decompose the phase-space $\mathcal{Y}$ of A into a finite set of equivalence disjoint classes, $G_1\ G_2 \dots G_g$ such that

$$\mathcal{Y} = G_1 \cup G_2 \cup G_3 \cup \dots \cup G_g \tag{19}$$

$$G_\alpha \cap G_\beta = 0 \quad \text{if} \quad \alpha \neq \beta$$

The decomposition or partition induced by the connection relation c, being exhaustive and exclusive: each state belongs to some G_α for one and only one value of α.

Let now $G \subset \mathcal{Y}$ be any group of states, and $y_0 \in G$ an arbitrary state of G. Consider now the set $\sigma^{*} \lfloor y_0$ of successors of y_0. If $y_1 \neq y_0$ and $y_1 \in G$, is another state of G, because of the definition of G, y_1 will be a successor of y_0, that is

$$y_1 \in \sigma^{*} \lfloor y_0$$

or

$$\lfloor y_1 \subset \sigma^{*} \lfloor y_0$$

but then

$$\sigma^{*} \lfloor y_1 \subset \sigma^{*} (\sigma^{*} \lfloor y_0) = \sigma\sigma^{*} \lfloor y_0 = \sigma^{*} \lfloor y_0$$

$$\sigma^{*} \lfloor y_1 \subset \sigma^{*} \lfloor y_0 \tag{a}$$

Repeating the argument, starting with y_1, we can similarly show that

$$\sigma^{*} \lfloor y_0 \subset \sigma^{*} \lfloor y_1 \tag{b}$$

From a) and b) we derive

$$\sigma'' \lfloor y_0 = \sigma'' \lfloor y_1 \qquad \text{(b)}$$

that is, any two states of G have the same set of successors.

Again, the whole argument can be applied to $\breve{\sigma}$ and we can conclude that, for any two elements $y_1, y_2 \in G$,

$$\sigma'' \lfloor y_1 = \sigma'' \lfloor y_2 \qquad\qquad \breve{\sigma}'' \lfloor y_1 = \breve{\sigma}'' \lfloor y_2 \qquad \text{(20)}$$

But since the group G itself is the union of the unitary classes of its elements, the set of all successors of all the elements of G, that is, the set $\sigma'' G$ will be the union of the sets of all successors of the elements of G. By (20), all these sets being equal, their union is merely the set of successors of any of them. The same can be said about precessors. The conclusion is then, that for every state $y \in G$ the successors of y are the $\sigma'' G$ and the precessors of y are the $\breve{\sigma}'' G$, in symbols:

$$(y)\,(y \in G \,.\supset : \sigma'' \lfloor y = \sigma'' G \therefore \breve{\sigma}'' \lfloor y = \breve{\sigma}'' G) \qquad \text{(21)}$$

Consider now the sequents of G: $s'' G$. This is a set of states completely determined by G itself and by the sequence relation. There are two alternatives: either the set $s'' G$ is contained in G in which case all sequents of G are G's themselves, or some sequents of the G's will be outside the group G. By using the complement G' of G, namely, the set of states not contained in G, the two alternatives can be formulated as:

$$\text{(A)} \quad G' \cap s'' G = \Lambda$$

$$\text{(B)} \quad G' \cap s'' G \neq \Lambda$$

Case (A) will occur, if and only if, the $s^* G \subset G$. This, however, implies that for any l, $s^{l*} G \subset G$ and thus, $\sigma^* G \subset G$. But since for every group G it is always true that $G \subset \sigma^* G$ we should have $G = \sigma^* G$.

That is, the set of all the successors of the elements of the group is the group itself. Such a group will be called a *final group*. Clearly, a necessary and sufficient condition for G to be a final group, or in other words, for the fulfillment of the condition $G = \sigma^* G$ is, that $s^* G \subset G$.

A final group can be characterized as a hereditary property of the states of A under the relation of sequence and, consequently, under the relation of succession. Both relations transmit the final group character.

Case (B) occurs if some $y \in G$ possesses a sequent outside G. Such a group will be called a *transient group*.

It can easily be seen, that a necessary and sufficient condition for a group G to be transient is, that some state of G possesses a sequent not belonging to G. In symbols

$$(\exists y)(\exists y^1)(y \in G \,.\, y^1 \bar{\in} G \,.\, s(y^1, y)) \qquad (22)$$

Suppose that y_1 is a state of G having a successor not contained in G. If y_2 is any other state of G, since G is a group, both, y_1 and y_2, have the same set of successors. Consequently, y_2 will have also a successor out of G. In other words, in a transient group each state y of the group has at least a successor outside the group:

$$(y) \quad y \in G \,.\, \supset . \,(\exists y^1)(\sigma(y^1, y) \,.\, y^1 \bar{\in} G) \qquad (23)$$

The totality $\mathcal{Y}$ of states of A can thus be decomposed in a set of groups, the states of each group interconnected among them, and the groups themselves can be classified as final or transient. Whenever the automaton enters a final group, it will remain within the group for all future times: This is the reason for the name final. In a transient group, however, the automaton may leave the group through a proper combination and timing of input symbols.

It can be easily shown, that every automaton A, possesses at least one final group. The proof is as follows:

If $\mathcal{G} = G$ consists of a single group, this group is evidently final and the assertion proved.

If $\mathcal{G} = G_1 \cup G_2 \cup \ldots\ldots \cup G_q$ contains a number $q > 1$ of groups, consider G_1. If it is final, the assertion is proved. If it is not, it will be transient and there exists at least one successor of G_1 outside G_1. This successor must belong to some of the remaining groups, G_2 say. If G_2 is final, the assertion is proved. It is transient, it will have a successor outside G_2. This succesor cannot belong to G_1, because if it were, all elements of G_1 would be successors of the elements of G_2, and since the elements of G_2 are themselves successors of the elements of G_1, the states of G_1 and G_2 would be connected, contradicting the assumption that they belong to two different groups. The assumed successor must then belong to a group different from G_1 and G_2, say G_3. And the argument can be repeated anew. By induction, since the number of groups is finite, one will eventually arrive at some group, all of whose sequents belong to the group, that is, one will find at least one final group.

These arguments can be reproduced using instead of s and σ the inverse relations $\breve{s}$ and $\breve{\sigma}$. In this manner, considering the set $\breve{s}"G$ of precessors of the G's, there are two alternatives:

$$(A_1) \quad G' \cap \breve{s}"G = 0$$

$$(B_1) \quad G' \cap \breve{s}"G \neq 0$$

In case A_1) $\breve{\sigma}"G = G$. All precessors of the G's will be G's themselves. A group G having this property will be called an *initial group*. A necessary and sufficient condition for a group to be initial is that $\breve{s}"G \subset G$.

In case B_1) G will be called a *non-initial group.* Such a group has the property that all its states have a precedent not belonging to the group. However, in order to show the non-initial character of the group, it is enough to exhibit a single state of G having a precedent outside G.

Just as with the direct relation, one can show that every automaton A possesses at least one initial group.

A group, which is at the same time initial and final, will be called an *isolated group.* If $\mathfrak{Y}$ consists of a single group, this is, trivially, isolated.

Although the automaton will always possess at least an initial and at least a final group, it may not have any isolated group, except in the just mencioned trivial case.

9. Inner Organization of automata.

For the purpose of the present discussion, it is convenient to imagine a group diagram of A, obtained from the kinematic diagram as follows:

For each group G_α draw a line enclosing all the states of G_α. If this is done for all groups, the kinematic diagram will be divided in q regions, each region corresponding to a group. In this diagram, final groups will be characterized by the fact, that no lines leave the corresponding region. Initial groups will correspond to regions having only outgoing lines. Isolated groups will have neither incoming nor outgoing lines.

We shall now show, that the relations of sequences succession and their inverses, induce corresponding relations among the groups themselves.

Let G be a transient group. We know that there exists a state y_1 belonging to the succession of G but outside G: $y_1 \in \sigma^* G$. $y_1 \bar{\in} G$. y_1 then, will belong to some other group $G_1 \neq G$. y_1 is a successor of every state of G. Since it belongs to G_1, every state of G_1 will in turn be a successor of y_1, hence, a successor of every state of G.

Groups have then the property that, if one element of a group is a successor of one element of another group, every element of the first group will be a successor of every element of the second. It is natural to say then, that the second group, as a whole, is a successor of the first, introducing in this manner a relation of succession among groups. This relation we call Σ and define by

$$\Sigma(G_1, G) = (y_1)(y)(y_1 \epsilon G_1 . y \epsilon G : \supset . \sigma(y_1, y)) \qquad (24)$$

A weaker but equivalent definition would be

$$\Sigma(G_1, G) = (\exists y_1)(\exists y)(y_1 \epsilon G_1 . y \epsilon G : \supset . \sigma(y_1, y)) \qquad (24a)$$

We can think of Σ as a relation defined over the *group-space* of A, understanding by this, the set whose elements are the groups of states of A.

It is possible to define also a relation of sequence among groups, understanding sequence as immediate succession, by saying, that G_1 will be a sequent of G, $S(G_1 G)$, if there is in G_1 a sequent of some state of G.

$$S(G_1 G) = (\exists y_1)(\exists y)(y_1 \epsilon G_1 . y \epsilon G . s(y_1, y)) \qquad (25)$$

Of course, in this case, every state of G_1 will be successor of every element of G, but the fact that one state of G_1 is a sequent of one state of G, causes this relation to be of immediate character. Whenever this happens, we can substitute the set of individual arrows in the group diagram, by a single arrow, going from G to G_1. (Fig. 10)

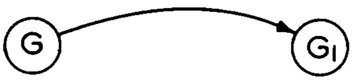


Fig.10

Just as with states, once a sequential relation S has been defined, one can introduce sequences of various orders by means of powers S^l, and the ancestrals $\natural S$, $\# S$, the last, meaning in this case, the succession Σ :

$$\Sigma = \# S = I \cup S \cup S^2 \cup S^3 \cup \ldots\ldots = \bigcup_{l=0}^{\infty} S^l \qquad (26)$$

In spite of these similarities, there exists an important difference between the ways in which states and groups can be interrelated. We shall presently show that, among groups, there is no such a thing as a connection relation. On the contrary, if $G \neq G_1$ are two groups and $\Sigma(G_1, G)$ is true, that is, if G_1 is a successor of G, then G cannot be a successor of G_1 :

$$\Sigma(G_1, G) \supset . \sim \Sigma(G, G_1) \qquad (27)$$

The reason is quite simple. If G_1 is a successor of G, every state $y_1 \in G_1$ is a successor of every state $y \in G$. If besides, G were a successor of G_1, the states of G would in turn be successors of the states of G_1. Then, every state of G would be connected to every state of G_1, and that would be possible only if $G = G_1$, were the same group, but not different groups, as assumed.

Graphically, this means that in the group diagram of A the relational arrows can never form closed paths. Sequential relations among groups show then a sort of irreversible character. Once the automaton leaves a group, it will never enter the group again.

We shall try now to establish the most general type of relation existing among groups and determine their organization.

For this purpose, we pick a certain group G, and look for groups that might be possibly related to it by Σ or $\breve{\Sigma}$, within the limitation imposed by equation (27) which can be conveniently written as :

$$\Sigma \subset \breve{\Sigma}' \qquad\qquad \breve{\Sigma} \subset \Sigma' \tag{27a}$$

or

$$\Sigma \cap \breve{\Sigma} = 0 \tag{27b}$$

with identical properties for S.

Starting from our selected G, one may have sequents S, or precedents $\breve{S}$ which, together, can be considered as the first degree relatives of G. As a whole, these are denoted by the relation $S \cup \breve{S}$. Next, one may have second degree relatives, namely S^2 sequents, $\breve{S}^2$ precedents, sequents of precedents $S\breve{S}$ and precedents of sequents $\breve{S}S$. All of these are related to G, by the second degree relation

$$S^2 \cup S\breve{S} \cup \breve{S}S \cup \breve{S}^2 = (S \cup \breve{S})^2$$

Next, one could consider third degree relatives, all of which would hold with G the relation

$$S^3 \cup S^2\breve{S} \cup S\breve{S}S \cup S\breve{S}^2 \cup \breve{S}S^2 \cup \breve{S}S\breve{S} \cup \breve{S}^2 S \cup \breve{S}^3 = (S \cup \breve{S})^3$$

Proceeding in this manner, one can define for all $k = 1, 2, 3, \ldots$ relatives of degree k by the relation $(S \cup \breve{S})^k$, which we want to extend by introducing the 0'th power $(S \cup \breve{S})^0 = I$, meaning that we consider any group as 0'th degree relative of itself. The relations

$$F_n = \bigcup_{k=0}^{n} (S \cup \breve{S})^k \tag{28}$$

include all relatives up to degree n, and

$$F = \bigcup_{k=0}^{\infty} (S \cup \breve{S})^{k} \tag{29}$$

contains relatives of any degree.

Remembering (26) it is not difficult to see that by a mere rearrangement of terms, (29) can be written in the equivalent form

$$F = \bigcup_{k=0}^{\infty} (\Sigma \cup \breve{\Sigma})^{k} = (\Sigma \cup \breve{\Sigma})^{\infty} \tag{29a}$$

We illustrate a few connections of the sort being discussed, (Fig. 11).

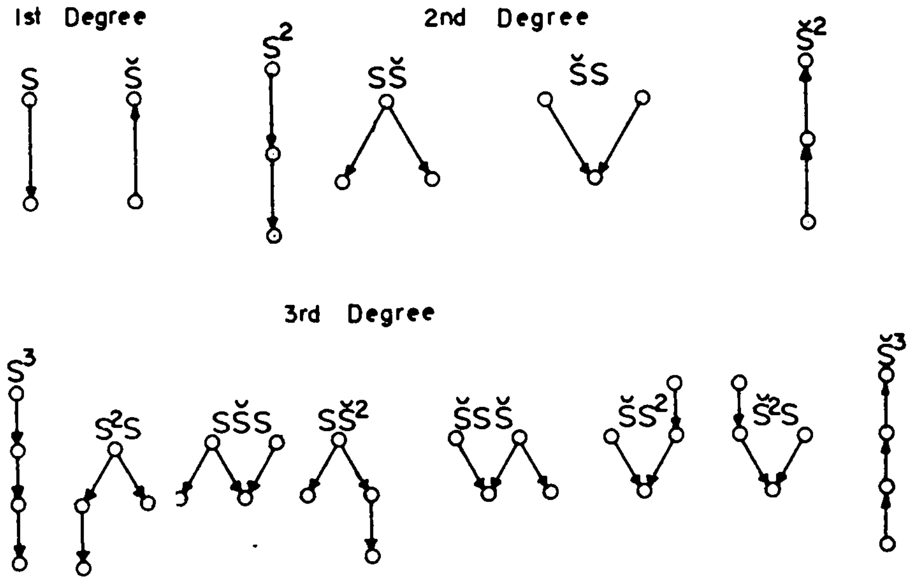


Fig.11

The type of organization one encounters among groups, resemble very closely that normally found in living populations, when their individuals are examined by their ancestry and descendence, in order to be grouped in families, according to their genealogic tree.

In fact, the type of graph one obtains when no closed paths are allowed, is known in topology as *tree-like* graph and, when connected, a *tree*.

That the relation just introduced allows a classification of the groups of A into families, follows immediately from the fact that F is

$$\begin{array}{ll} \text{reflexive} & I \subset F \\ \text{symmetric} & F = \overset{()}{F} \\ \text{transitive} & F^2 \subset F \end{array} \tag{29b}$$

that is, an identity relation, which, as all relations of its kind, induces a partition of the elements of the set (groups this time), into exhaustive and disjoint equivalence classes. These can be formed by the usual procedure: One starts with some group G_1, say, and considers the family $A_1 = A(G_1)$ of relatives of G_1. If this family exhausts the whole set, this will consist of a single family. If it does not, one takes a 2nd group $G_2 \bar{\epsilon} A_1$, and forms its corresponding family $A_2 = A(G_2)$, etc. Eventually one arrives at a partition of the group-space of A into a finite number of disjoint subsets, $A_1\ A_2 \ldots A_r$.

Designating the group-space by the same letter A already used for the designation of the automaton itself, our decomposition can be expressed as

$$\begin{array}{l} A = A_1 \cup A_2 \cup \ldots \cup A_r \\ A_\mu \cap A_v = 0 \quad \text{if} \quad \mu \neq v \end{array} \tag{30}$$

Two groups belong to the same set A_v if and only if, they are relatives.

Groups falling in different trees have no connection whatsoever and will be said to be *extranious*. The automaton decomposes into a set of logically independent entities, each having the characteristics of a complete automaton. These entities we shall call *subautomata*.

We can apply now the same arguments employed this far for the analysis of A, to any of its subautomata, A_v, each being organized as already shown, except that it consists of a single tree. In particular we can assert, that each A_v possesses at least some initial group, a "Adam" of the generation, and at least a final group. The most general type of A_v is one containing several *initial groups*, transient, non-initial groups which will be called *inner groups* or *passing groups* and some *final groups*. The whole automaton can be formed by one or more trees of this type. Very often it may happen, that the tree contains only an initial group, and a single final group, and it may even reduce to one single group, in which case this group is *isolated*. Again, it often happens that a group consists of a single state.

This picture of the organization of A, throws some light on the nature of its expected behaviour. The reader will certainly be aware, that the notion of group depends on that of connection which we used in the sense that Moore[5] would call "strong connection". Our relations of sequence and succession which in this fundamental work would be called "weak connections", were not considered as connecting relations, because the word itself "connect" has a cognotion of reciprocity which as far as the behaviour of the automaton is concerned, corresponds to some reversibility. As long as A stays in a group, by a proper choice of input symbols, it can leave and return to any of the states of the group. This is the reason why we considered these states as connected. When a mere succession or sequential relation exists, there is an obvious irreversibility in the behaviour of A. Once a group is abandoned, A will never return again to it. We cannot understand this sort of behaviour as connected.

Loosely speaking, each group of states represents some mode of repetitive behaviour. As long as A remains in the group, repeated identical situations can arise, and A will show exactly the same type of behaviour under the same sequences of stimuli. Some event, however, may cause A to abandon the group. From this moment on, the mode of behaviour will change forever and will belong to a new type, that characterized by the new group. Eventually, A will attain one of its final groups.

From there on, its behaviour and the type of operations it can realize, will remain always the same. A set of stimuli will produce the same type of response. (no matter how complicated, and independently of the number of detailed responses). One can expect at most a finite set of different types of finite behaviour, the particular one actually attained depending on the past history of A in relation to the sequence of external stimuli.

As to the decomposition of A in two or more families, it is well known that as far as behaviour is concerned, this situation represents a set of disjoint possibilities. If A starts its life in an initial state belonging to some tree, it will remain forever in that tree, and show the type of behaviour corresponding to it. The remaining trees appear then as situations that might have possibly occurred, were it not by the fact that they did not. The reader will be aware by now, that we departed somewhat from the terminology sometimes used, in which an automaton or "machine" is defined through the phase-space and *any* subset of states is called a "submachine". The reason for this departure, is to conform to the universal and accepted usage of prefix "sub". as a qualifier of names for mathematical systems. Thus, the terms subgroup, subring, subfield, etc., designate not *any* subset of a group, a ring, a field, etc., but *certain* subsets, having themselves the properties of the whole. This is precisely the case for which we reserved the term "subautomaton", namely, a subset of the totality of states, representing an entity having the properties of a complete automaton, except for the fact, that its behaviour can be considered to belong to a single kind.

10. The Structure of Final Groups.

From the fact that a final group G is a set of connected states, some consequences can be derived as to the nature of its possible structure, and the mode of behaviour it represents.

The type of analysis we have in mind, is the one that is used in the theory of stochastic processes, in connection with Markov chains. If one substitutes "sequence relation" by "transition probability $\neq 0$" the analysis of Markov chains can be carried over, literaly, word by word, to the present case. Thus, we shall give the results, referring the reader to the literature[4] for details.

Each final group G is characterized by a positive integer $m \geq 1$ such that the states of G can be divided in m classes $g_1, g_2, \ldots\ldots g_m$ having the property that every state of g_k ($k = 1, 2 \ldots m$) has a successor in g_{k+1}. This is symbolically represented in Fig. 12.

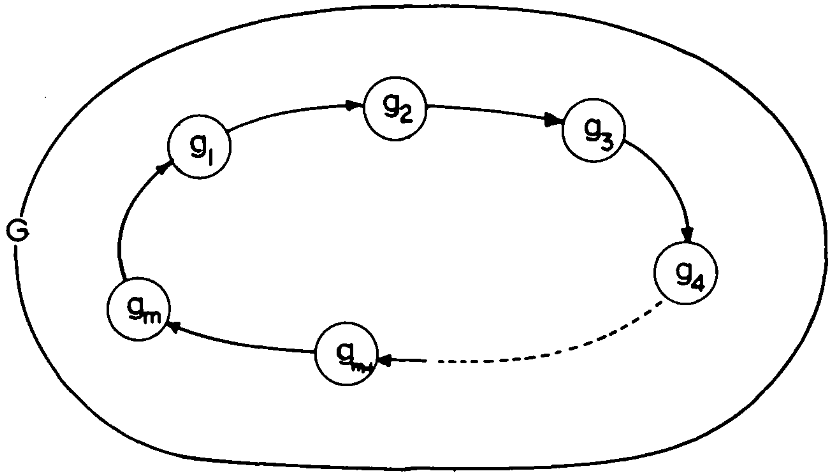


Fig.12

The relation of sequence leads then the automaton cyclically around cycles of m values. The g_k will be called *simple sets*. A group G having m simple sets, will be said to be cyclic with period m.

The mode of behaviour corresponding to a cyclic group, can be pictured as some sort of periodic behaviour with multiple choice in the sense that at each state, say of g_k, there is a number of alternatives whereby A according to the stimulus received, can go to one of the states of g_{k+1}. After a number of m steps, A will be found again in the simple set of departure. Hence, one must expect a cyclic behaviour with period m, capable of showing different "forms" but finite in number. In case, each simple set reduces to one state, the behaviour will be simply periodic.

An important particular case arises when $m = 1$. G then reduces to a simple set, in which case it is called ***simple group***. The behaviour will show some sort of repetitive character, but without any intrinsic periodicity.

11. Relation between the Automaton and the Medium.

It is not our purpose to examine in detail these matters for the moment, but merely to point out certain general principles closely connected with the relational formalism, laying the foundations for subsequent work to be presented elsewhere.

In the first place, we want to call the attention of the reader to the fact that everything this far said about the way in which states appear organized, depends on the input alphabet. Strictly speaking, when considering organization of the automaton, one should really say "organization relative to an input alphabet". It is clear that, if one assumes that for some reason, some symbols of the original alphabet will never be present, one might examine the resulting structure of the automaton under a sub-alphabet. The structure appears naturally modified in an obvious sense. Some relations, originally present, will disappear, certain connections may be broken and, in general, the "degree" of organization will be lessened. The general tendency will be to decompose original units, such as groups and families, in sets of smaller units. This general property, as is well known, can be practically applied to the design of automata planned for several different operations. Each such operation is handled through an adequate sub-alphabet through a particular organization of states. Changes from one operation to another can be effected by a change of sub-alphabet as determined by a set of proper control signals.

The overt behaviour of the automaton depends not only on its output alphabet, but on the particular output symbols, as determined by its inner states. This is the content of the second equation $z = g(y)$ [pag. 22, (1)], which assigns one and only one output symbol to each inner state y. In general, there may be several states providing the same symbol.

In line with the logical formalism, output symbols can be considered as properties of states. To each state y, one can assign a property z, namely, that of delivering a certain symbol, as a result of A being in state y. One may say that state y has the property z, and write, according to the usual formulation of predicate calculus, the formula $z(y)$, which could be read as "y is z".

Each output symbol merely becomes a class of states, namely,

$$z = \hat{y}\, z(y) \tag{33}$$

In the next report of this series, we shall see that this apparently trivial manner of considering outputs, has some serious consequences. States can be separately considered with respect to two properties: external stimuli-induced transitions and overt behaviour, dependent on output properties. These, however, are by no means intrinsic properties of the states but represent a conventional coding which fixes a possible kind of overt behaviour. We shall return later to this matter.

12. Endomorphisms and Automorphisms in Automata.

For future reference, we want to apply the relational technique to the consideration of inner transformations in the phase-space of an automaton.

An inner transformation t, is a rule that assigns to each $y \in \mathfrak{Y}$, a transformed, or image, $\eta \in \mathfrak{Y}$ the transformed of y.

$$t : y \to \eta = t' y \tag{1}$$

Each original state has, under t, one and only one image, $\eta = t'y$, but several states may have the same image. This transformation can be interpreted as a relation $t(\eta, y)$ between the original states and their images, as expressed by the statement "η is the t of y". The inverse relation $\breve{t}(y, \eta)$ meaning "y has η as a t", allows a description of the classes

$$\breve{t}'' \lfloor \eta = \hat{y}\, \breve{t}(y, \eta) = \hat{y}\, t(\eta, y) \tag{2}$$

$\breve{t}'' \lfloor \eta$ is the class of all y's transformed into the same η. Any two of these, $y_1, y_2 \in \breve{t}'' \lfloor \eta$ are both $\breve{t}$ of the same η, or, equivalently, one is a $\breve{t}$ of a t of the other, that is, each is related by $\breve{t}t$ to the other.

It is easily seen that this form of partnership-relation is

reflexive	$I \subset \breve{t}t$
symmetric	$(\mathring{t}t)^{\cup} = \breve{t}t$
transitive	$(\breve{t}t)^2 \subset \breve{t}t$

and, as an identity relation, effects a partition of the states in equivalence-classes, each class being that of all states having the same image. These classes are disjoint

$$\breve{t}'' \lfloor \eta_1 \cap \breve{t}'' \lfloor \eta_2 = 0 \quad \text{if} \quad \eta_1 = \eta_2 \tag{3}$$

if $H = \{\eta\}$ is the set of all images, it is clear that H will be the projection of $\mathfrak{Y}$ by t:

$$H = \hat{\eta}(\exists y)\, t(\eta, y) = t'' \mathfrak{Y} \tag{4}$$

Since every image $\eta \in \mathcal{Y}$ is also an element of $\mathcal{Y}$, the transformation determined in $\mathcal{Y}$ by each input symbol x, induces a corresponding transformation ξ in H and its associated relation. Since for a given t the relation ξ in H is uniquely determined by x, we shall use for it the same name x, both for the symbol and for the relation.

Each η will have a x-sequent $x'\eta = x'(t'y)$.

A transformation (or relation) t that fulfills the condition

$$(x)(y)\big(x'(t'y) = t'(x'y)\big) \tag{5}$$

is called an endomorphism.

If t is an endomorphism, (5), or the equivalent expression

$$(x)(y)\big(xt'y = tx'y\big) \tag{5a}$$

are satisfied. The sketch shown in figure 13 illustrates this property.

$y_1 = x'y \qquad \eta = t'y$

$\eta_1 = t'y_1 = tx'y =$

$= x'\eta = xt'y$

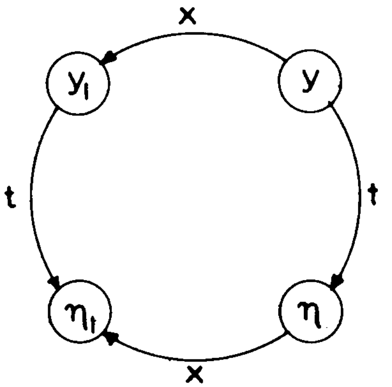


Fig.13

Condition (5) can be fullfilled only if

$$(x)\,(xt = tx) \tag{6}$$

namely, if t commutes with every x.

Remembering that in general the relational product is not commutative, we shall define the *commutator* $[x, t]$ of two relations x and t, as the symmetric difference or disjunction

$$[x, t] = xt \;\Lambda\; tx \tag{7}$$

According to this definition, a commutator is in turn a relation and thus, a class of ordered pairs which, according to (7) contains the x's of the t's that are not t's of the x's and the t's of the x's that are not x's of the t's. Clearly, two relations commute, if and only if, their commutator vanishes. Equation (6), can then be written as

$$(x)\;[x, t] = \vartheta \tag{8}$$

A relation such as t commuting with all x's, will give rise to properties of states and of the phase space itself, which do not change, in the sense of remaining the same, when the automaton undergoes any series of transitions produced by the alphabet of external stimuli. By this reason, a t fullfilling (8) or any of its equivalents, will be said to be a *constant of motion* of the automaton.

Because of this character, we can easily show that t commutes with sequence and succession relations:

a.- $[s, t] = \vartheta$

b.- $[s^l, t] = \vartheta \quad (l = 0, 1, 2 \ldots)$

c.- $[\natural s, t] = \vartheta$

d.- $[\# s, t] = [\sigma, t] = \vartheta$ (9)

a) follows immediately from the fact that $s = \bigcup_k x^k$ because then

$$st = \bigcup_k x_k t = \bigcup_k t x_k = t \bigcup_k x_k = ts$$

b) can then be readily proved by induction, since we know that $st = ts$ and by assuming that $s^k t = ts^k$, then,

$$s^{k+1} t = ss^k t = sts^k = tss^k = ts^{k+1}$$

which shows the commutation to be true for all values of l. The case $l = 0$ is trivially true, since $s^0 = I$, and the identity commutes with every relation.

c) then follows from the fact that $\natural s = \bigcup_l s^l$ and

d) is immediate because of the trivial commutativity of I.

By taking the inverse of (6), we can show that $\breve{x}$ and $\breve{t}$ commute. In fact, whenever the commutator of two relations vanishes, the commutator of their inverses will also vanish. From (8) we can infer

$$[\breve{x}, \breve{t}] = \vartheta \quad (8a)$$

and similar expressions for the inverses of (9). However, because of the fact that both x and t are many-to-one relations, the inverse of one does not commute with the direct form of the other. It can be easily shown that in this case, instead of the equality appearing in (6), one has, instead, an inclusion relation, namely,

$$t\breve{x} \subset \breve{x}t$$

$$x\breve{t} \subset \breve{t}x \tag{10}$$

From (6) and (10) one readily obtains:

$$x\breve{t}t \subset \breve{t}tx \tag{11}$$

Now, $\breve{t}t$ is that relation characterizing the set of states possessing the same image. Hence, according to (11), the x-transform of a set of t-partners is always contained in the set of t-partners of the x-transforms. This property completely characterizes endomorphisms and can be used as the basis for a systematic procedure leading to the determination of all endomorphic images of a given automaton, to be treated later.

Because of the identical transformation properties of the states contained in each set, the $\breve{t}" \lfloor \eta$ will be called sets of *covariant states*.

Under an endomorphism t, the set of images $t" G$ of the elements of a group G, must be contained in a group Γ of the endomorph:

$$t" G \subset \Gamma \tag{12}$$

Then, every group Γ of the endomorph, is the image of one or more original groups. This means however, that t establishes a many-to-one correspondence

$$t : G \rightarrow \Gamma = T' G \tag{13}$$

from the original group space to the group space of the endomorph. The transformation induced by t in group space, we denote by T and its associated relation by $T(\Gamma, G)$. Endomorphic mappings have then the property that, each group Γ of the endomorph, is the image of a set of complete groups. The sets $\breve{T}'' \lfloor \Gamma$ are classes of groups projected by t onto the same group Γ. The reader should notice that again, as with transitions induced by external stimuli, groups behave as units, which could be called of *second order*, if states are interpreted as being of the *first order*. The concept of group of states, seems to be of a very fundamental nature, and appears as the simplest mode of organization possessing the properties of an entity. In fact, (13) can be interpreted by saying that, under the set of endomorphisms, the concept of group is covariant.

Identical considerations concerning families of states, or, as we also call them, sub-automata, show that under an endomorphism t, the images $t''A$ of any *sub-automaton A must be contained in some sub-automaton of the image:*

$$t''A \subset \mathcal{A} \tag{14}$$

Again, families of states appear as *third order* units of organization among states, and represent what we could call the highest coherent type. Higher order units, in so far as the original input alphabet is fixed, appear as mere agregates or "colonies" of families, but not as organizations possessing a definite inner structure. It is interesting to observe, that the type of behaviour determined by a set of canonical equations, allows only two degrees of organized behaviour: first, what we called *mode*, as characterized by groups, and a second which could be called *type*, corresponding to a family. Modes are determined by an inter-connective relation, whereas types correspond to a generation relation, giving rise to an organization of ancestry and decendence.

It appears that our logical formalism does not allow the conception of a complex operation beyond the type, as a unit. Such an operation would be immediately decomposed into an agregate of types, having no connection among them.

A final word will be said about another type of transformation which can be treated with the same formalism: these are the so-called *automorphisms*. First, one may consider a particular type, that of *inner automorphisms*. These are one-to-one endomorphisms, so that properties already discussed in connection with the latter kind, can be applied to the former. The main difference lays in the fact that an inner automorphism being a one-to-one transformation of phase space into itself, practically effects a permutation of the set of states. Each inner automorphism p is associated to a permutation of states. As a relation, is a constant of motion and commutes with every x.

$$[x, p] = 0 \tag{15}$$

And thus, with sequence and connection relations.

This time, however, because of the bi-uniform character of the relation, the inverse $\breve{p}$ is also a commutator of all x's

$$[x, \breve{p}] = 0 \tag{16}$$

And hence, with inverse succession and sequence relations. In particular, every p will commute with connective and genealogic relations:

$$[c, p] = [\sigma \cup \breve{\sigma}, p] = 0 \tag{17}$$

Under an inner automorphism p, the automaton remains invariant. The transformed automaton is formally identical in every respect to the original, not only as far as the structure of its inner relations, but even as to names of the states. In short, one can characterize automorphisms as transformations leaving all properties of automata invariant: groups go over into groups, families to families, etc.

The one-to-one character of inner automorphisms, allows the interpretation of the inverse $\breve{p}$ of any p, not only as a relational inverse, but also, as a formal inverse in the sense of the theory of transformations. Relational operations such as products and inverses, acquire now the usual meaning, and can be dealt with, according to the well known transformation theory.

In particular, considering that the identical I relation is trivially an inner automorphism, that to any automorphism p there corresponds a unique inverse $\breve{p}$ such that

$$p\breve{p} = \breve{p}p = I \tag{18}$$

and that the product (associative) of automorphisms is again an automorphism,one sees that the set $\mathfrak{P} = \{p\}$ of inner automorphisms of an automaton, forms a group. Inner automorphisms, accordingly, represent inner symmetries of the automaton which determine a particular form of equivalence among its states. In fact, states that can be transformed in each other by an automorphism of $\mathfrak{P}$ are *indistinguishable* in the sense of Moore[5]. More important is the fact that the operations associated with automorphically equivalent states, are themselves indistinguishable.

A wider class of automorphisms is obtained, when one performs a joined transformation of phase space and of the input alphabet, requiring that, as a result of the transformation the whole system remains invariant. We could call ***extended automorphism a***, a one-to-one correspondence

$$\begin{aligned} x &\to \bar{x} = a'x \\ y &\to \bar{y} = a'y \end{aligned} \tag{19}$$

leaving the whole system invariant. The previously discussed inner automorphisms are particular cases of (19) arising when that part of a effecting the input symbols, becomes an identity.

ACKNOWLEDGEMENT

The writer wishes to express his indebtedness and gratitude to his distinguished colleague Mrs. Gertrude Kurz De Lara for her kind cooperation and constructive criticism in the writing of the manuscript.

REFERENCES

1. W.v.O. Quine, "Mathematical Loqic", New York, 1940.
2. H. Reichenbach, "Elements of Symbolic Logic", New York, 1947.
3. W. Ross Ashby, "Introducción a la Cibernética", Buenos Aires, 1960.
4. A. Blanc-Lapierre and R. Fortet: "Théorie des Fonctions Aléatoires", Paris, 1953.
5. Shannon and McCarthy, (eds.) "Automata Studies", Princeton University Press, 1956.