INFORMATION AND CONTROL 14, 299–317 (1969)

Tessellation Automata

H. YAMADA[1]

The Moore School of Electrical Engineering, University of Pennsylvania, Philadelphia Pennsylvania 19104

AND

S. AMOROSO

US Army Electronics Command, Fort Monmouth, N.J. and Stevens Institute of Technology

Certain mathematical studies of pattern recognition, evolution theories, and self-reproducing automata motivated the definition of the tessellation automaton which is a mathematical model of an infinite array of uniformly interconnected identical finite-state machines. Each machine is capable of changing state at discrete time steps as a function of the states of other machines in the array and inputs that act as environmental changes to the array. This model can embed all the models used in the studies mentioned above, and may serve as a unifying framework.

In this first paper certain basic properties of tessellation automata are developed that are intended to serve as an introduction to the more specific studies to be reported in the sequel.

I. INTRODUCTION

Most of the current work in abstract machine theory is concerned with models that can naturally be considered only as single machines and not as arrays of intercommunicating machines. In order to provide an adequate framework in which to study theories of certain phenomena such as pattern recognition and some important biological processes, there is some evidence to indicate that an extended machine concept would be useful. After some initial work, J. von Neumann (1966) found a two-dimensional array of identical finite-state machines to be a con-

[1] The first author was partially supported by the National Institute of Health (under Contract No. 5 R01 GM-13, 494-03), the National Science Foundation (under NSF Grant GP-5661), and the US Army Research Office (Durham) (under Contracts No. DA-31-124 ARO(D)-97 and No. DA-31-124 ARO(D)-98 MOD. #5).

venient model in which to study self-reproducing automata. S. H. Unger (1958, 1959) found that a similar two-dimensional array of finite-state machines had some interesting processing capabilities for pattern recognition tasks, and D. L. Slotnick (1962) has applied such arrays to numerical computations. N. A. Barricelli (1957, 1963) has shown, in what is essentially a one-dimensional array, some state-pattern phenomena that can be interpreted as analogues of evolutionary processes and natural selection, and C. Y. Lee (1962, 1963) has indicated how a one-dimensional array might be useful for information retrieval systems.

Each of the above efforts used arrays of machines operating in much the same way, but one should note that each was concerned with a particular application and not with the theoretical capabilities and limitations of the machine arrays themselves. E. F. Moore (1962) (see also J. Myhill (1963)) appears to be the first to have attempted a study of the machine concept itself. In this work he defined a class of machines that he called *tessellation structures*, proved some of their properties, and outlined some directions for further study.

Somewhat independent of the above mentioned efforts, motivated in part by the recent advances in batch fabrication and the integrated circuit technology, is a great deal of switching theoretic work concerned with organizing complex logical systems based on uniform arrays of switching elements. A recent survey paper by R. C. Minnick (1967) summarizes this work.

The development of a comprehensive framework in which a general theory of logical arrays could be studied would be most desirable. Such a general theory would hopefully contribute to a better understanding of the applications of these arrays. The present paper is the first report on our attempt to scrutinize tessellation structures from a fairly general point of view without regard to any immediate applications. We do not claim that this work is the unifying framework for logical arrays; rather it is more a continuation of the study of the structures that Moore defined. It is our hope, however, that our efforts will contribute to the development of a unified theory.

II. THE TESSELLATION AUTOMATON

The intuitive picture of what we hope to formalize by the tessellation automaton is that of an infinite regular array of identical finite-state machines, where each machine can receive information directly from only a finite number of neighboring machines, and where each machine is connected to its neighbors in a uniform way throughout the array. Each machine can change its state only at discrete time steps as a function of the states of the machines in the uniformly defined finite set of neighboring machines. *This function may be changed from time step to time step, but it is identical for each machine at any given time step.*

Our formal definition of a tessellation automaton takes the form of a quadruple

$$(A, E^d, X, I)$$

where

1. A is a finite nonempty set called the *state alphabet*. This set corresponds to the state set of any one of the finite-state machines in the intuitive picture given above.

2. E^d, called the *tessellation array*, is the set of all d-tuples of integers. An element in E^3, e.g., can be visualized as the name of the machine situated at the lattice point in 3-space indicated by the triple. The elements of E^d will be referred to as *cells*, and we shall often abbreviate cell $(i_1, \cdots, i_d)$ by just i. We shall refer to d as the *tessellation dimension*.

3. X, called the *neighborhood index* of the tessellation automaton, is an n-tuple of *distinct* d-tuples of integers, where d is the tessellation dimension. X will be used to define a correspondence between cells and certain n-tuples of cells. If $X = (\xi_1, \cdots, \xi_n)$, $\xi_k = (x_{k_1}, \cdots, x_{k_d})$, $1 \leqq k \leqq n$, then $N(X, i)$, called the *neighborhood* of cell i, is the n-tuple $(i + \xi_1, \cdots, i + \xi_n)$, where $i + \xi_k$ is the componentwise sum of the d-tuples, i.e., $i + \xi_k = (i_1 + x_{k_1}, \cdots, i_d + x_{k_d})$. A cell j is called a *neighbor* of cell i if and only if $j = i + \xi_k$, for some component ξ_k of X. Since the neighborhood of each cell in the given tessellation automaton is defined from the same X, the "relative positions" of the neighboring cells with respect to any cell can be thought of as being the same throughout the array. The neighborhood of cell i corresponds to the finite set of machines in the intuitive model *from* which the machine at point i has direct connections.

Before we can complete the definition by specifying I, we need some preliminary concepts. Let c be an arbitrary mapping from E^d into A, and let C be the set of all such mappings. We shall refer to these mappings as (array) *configurations* for the given tessellation automaton. The image of $i \in E^d$ under c, $c(i)$, will be referred to as the *contents of cell i*, or the configuration of cell i, in configuration c. Note that if the cardinality of A, $\#(A)$, is at least 2, and if $d \geqq 1$, then C will be a nondenumerably infinite set. By the *configuration of the neighborhood* of cell i in configuration c we mean the n-tuple in A^n denoted by $c(N(X, i))$ and defined to be $(c(i + \xi_1), \cdots, c(i + \xi_n))$ where $X = (\xi_1, \cdots, \xi_n)$. Let σ be an arbitrary mapping from A^n into A (note n is the number of components in X), and let L be the set of all such mappings. These mappings in L will be referred to as *local transformations.* For each $\sigma \in L$ there is a uniquely determined mapping τ from C into C defined as follows. For any $c, c' \in C$, $\tau(c) = c'$ if and only if for each cell i, $c'(i) = \sigma(c(N(X, i)))$. In other words, τ is the mapping that transforms one array configuration to another whereby each next cell value is determined by the same local transformation σ operating on the present values of the neighboring cells. Let T be the maximal set of such transformations for a given A, E^d, and X, i.e., all possible local transformations are used. We shall refer to the elements of T as *parallel transformations,* and to T itself as the *total input alphabet* for the given automaton. We shall often write $\tau(c)$ as $c\tau$, $\tau_1(\tau_2(c))$ as $c\tau_2\tau_1$, etc.

4. We can now complete the definition of the tessellation automaton by defining I to be any nonempty subset of T. I can be thought of as specifying which of the possible next-state functions are actually "wired in" for the array of machines.

Note that for any finite set A and any positive integer d, there is a tessellation automaton (A, E^d, X, I). On the other hand, X must be a list of d-tuples, and I is limited by the choice of A and X.

Let $\Xi^{(d,\, n)}$ be the set of all n-tuples of distinct d-tuples of integers (i.e., the set of all possible neighborhood indices with n components for a tessellation dimension d), and let $\Sigma^{(A,\, d)}$ be the set of all mappings from C into C, where $C = \{c \mid c{:}E^d \to A\}$. If $L^{(A,\, n)}$ is the set of all mapping from A^n into A (local transformations), we can define a mapping δ from $L^{(A,\, n)} \times \Xi^{(d,\, n)}$ (properly) into $\Sigma^{(A,\, d)}$ by: For any $\sigma \in L^{(A,\, n)}$ and any $X \in \Xi^{(d,\, n)}$, $\delta(\sigma, X) = \tau$, where τ is the parallel transformation on C into C specified by σ and X. The range of δ operating on $L^{(A,\, n)} \times \Xi^{(d,\, n)}$ is the set $\mathfrak{T}$ of all possible parallel transformations for any tessellation automaton with state alphabet A, a tessellation dimension d, and a $\#(X) = d$. Fixing $X_j^{(d,\, n)} \in \Xi^{(d,\, n)}$, we see that $\delta(L^{(A,\, n)} \times \{X_j^{(d,\, n)}\}) = T$ (or $T^{(A,\, d,\, n)}$). Note that $\delta{:}L^{(A,\, n)} \times \{X_j^{(d,\, n)}\} \to T^{(A,\, d,\, n)}$ is one-to-one and onto.

We can define $\delta_{X_j^{(d,n)}}{:}L^{(A,\, n)} \to T^{(A,\, d,\, n)}$ and also its inverse

$(\delta x_j^{(d,n)})^{-1}(T^{(A,d,n)}) = L^{(A,n)}$. We denote $(\delta x_j^{(d,n)})^{-1}(I^{(A,d,n)})$ by $J^{(A,n)}$, $J^{(A,n)} \subseteq L^{(A,n)}$.

If we let ψ be the mapping from $C \times I \rightarrow C$ defined by: $\psi(c, \tau) = c'$ if and only if $c\tau = c'$, then we can present a specified tessellation automaton (A, E^d, X, I) in a way that parallels a finite-state automaton without initial state or output, i.e., we can speak of the tessellation automaton

$$(C, I, \psi).$$

Letting I^* be the set of all finite sequences of elements of I, we can extend the domain of ψ to $C \times I^*$ by:

$$\psi(c, \epsilon) = c$$

$$\psi(c, \tau x) = \psi(\psi(c, \tau), x),$$

where ϵ is the null sequence and $x \in I^*$.

With a tessellation automaton M viewed in this way, many properties of finite automata that do not depend on the finiteness of the state set can be translated to M, e.g.,

$$(\forall c \in C)(\forall x, y, z \in I^*)(cx = cy \Rightarrow cxz = cyz), \quad \text{etc.}$$

(See, e.g., M. A. Harrison (1965)). We shall not pursue this further at this time.

Clearly $\#(A)$ must be greater than one or else no action can take place in the array. A tessellation dimension of zero would result in the tessellation automaton degenerating to a single finite-state machine. We shall spend a great deal of time with the case $d = 1$ in the sequel. If X is the 0-tuple, then T will contain $\#(A)$ transformations, and each is such that every cell in the array is sent to the same state, the state being determined solely by the transformation.

The automaton that von Neumann used can now be described as a tessellation automaton with $\#(A) = 29$, tessellation dimension two, $X = ((0, 0), (-1, 0), (1, 0), (0, -1), (0, 1))$, and I a unit set.

Unger and Lee were concerned with tessellation automata of dimensions two and one, respectively, and both used input sets of cardinality greater than one. The input sequences were used as programs of instructions for their machines.

When a configuration is viewed as simulating microorganisms, as in Barricelli's work, then it is natural to think of I as the set of possible environments for the microorganisms, and to think of a sequence of inputs as a history of environmental change.

For the remainder of this paper, when we speak of an *automaton* we shall mean tessellation automaton.

III. NEIGHBORHOOD INDEX AND LAMINATIONS

We examine now some effects that the choice of neighborhood index will have on the structure of a tessellation automaton.

With respect to an arbitrary automaton (A, E^d, X, I), $X = (\xi_1, \cdots, \xi_n)$, if there is an argument position j, $1 \leqq j \leqq n$, such that for *each* local transformation σ defining some input transformation in I, $\sigma(y_1, \cdots, y_n)$ is always independent of its jth component, then we say that the jth n-tuple ξ_j in X defines a dummy neighbor for each cell. Obviously any such ξ_j can be eliminated from X and the local transformations adjusted so that the resulting automaton is behaviorally unaltered. Alternatively, we can consider only automata with no dummy neighbors by properly choosing the subsets I. We will call such automata *neighborhood reduced,* and we shall henceforth assume that all automata considered are neighborhood reduced.

For any $X = (\xi_1, \cdots, \xi_n)$ and any two cells i and j, if there is a ξ_k in X such that $j = i + \xi_k$ or $j = i - \xi_k$, then we say that cells i and j are *immediate neighborhood related,* and we denote this relation by R_N. We say two cells i and j are *neighborhood related,* if either $i = j$ or there exists a sequence of cells $k_0, k_1, \cdots, k_m$ $(m \geqq 1)$ such that $i = k_0$, $j = k_m$, and $k_q R_N k_{q+1}$ for all q, $1 \leqq q < m$. This latter relation we denote by R_N^*. Clearly this is an equivalence relation, and the partition determined, $E^d/R_N^* = \{\Lambda_0, \Lambda_1, \cdots\}$, we call a *lamination* of E^d. If $\#(E^d/R_N^*) > 1$, we say the automaton is *laminated,* and we refer to the equivalence classes as *laminal subarrays.*

It is well known that the set E^d with operator ring Z, the set of integers, forms a module under the operations of the usual (componentwise) sum of d-tuples and multiplication of a d-tuple by an integer. Since any d-tuple in E^d is uniquely expressible as a sum of multiples of a subset, e.g., the standard basis elements $e_1 = (1, 0, \cdots, 0)$, $e_2 = (0, 1, 0, \cdots, 0)$, etc., the module is free. Z is a principal ideal domain. If M is a free v-module with a basis of d elements and if v is a principal ideal domain, then any submodule of M is free and has a basis of k, $k \leqq d$, elements. (See, e.g., p. 78, Vol. 2 of N. Jacobson (1953)). In view of this, if we define Λ_0 for any given automaton to be the laminal subarray containing $0^d = (0, 0, \cdots, 0)$, then we have

PROPOSITION 1. *$\Lambda_0 = \{m_1\xi_1 + \cdots + m_n\xi_n \mid m_i \in Z,\ \xi_i$ a component of X, $1 \leqq i \leqq n\}$ is a free left Z-module of finite type.*

We shall refer to Λ_0 as the *laminal submodule* of E^d determined by X, and we shall often use the notation $\Lambda_0(X)$. We say Λ_0 is *nontrivial* if $\Lambda_0 \neq \{0^d\}$.

COROLLARY 1.1. *$M = (A, E^d, X, I)$ is not laminated if and only if X contains a basis of E^d among its components.*

COROLLARY 1.2. *The lamination E^d/R_N^* is the quotient module of E^d by submodule Λ_0*, i.e., E^d/Λ_0.

COROLLARY 1.3. *For any laminal submodules $\Lambda^{(1)}$ and $\Lambda^{(2)}$, if $\Lambda^{(2)} \subseteq \Lambda^{(1)}$, then $\#(E^d/\Lambda^{(1)})$ divides $\#(E^d/\Lambda^{(2)})$.*

EXAMPLE. In E^2, let $X = ((-1, 2), (-1, -2))$. Then $\Lambda_0 = \{(m, 2m \pm 4n) \mid m, n \in Z\}$. It is clear that $\Lambda_k = \{(m, 2m \pm 4n + k) \mid m, n \in Z\}$, $k = 1, 2, 3$. In particular, $(0, 0) \in \Lambda_0$, and for $\zeta_k = (0, k)$, $k = 0, 1, 2, 3$, $\Lambda_0 + \zeta_k = \Lambda_k$.

For any $X = (\xi_1, \cdots, \xi_n)$, we say that cell j can be *reached by a forward trace* from cell i if and only if there exists a (possibly null) sequence $\xi_{i_1}, \cdots, \xi_{i_m}$ of elements from X such that $i + \xi_{i_1} + \cdots + \xi_{i_m} = j$. We denote this, in general nonsymmetric, relation by $\overset{*}{\Rightarrow}_N$. This definition is motivated by the fact that the content of cell i is a function of the content of cell j by an input sequence of length m only if cell j can be reached from cell i by a forward trace. The following is an easily verified necessary and sufficient condition that the contents of any cell i can influence the contents of any cell j.

PROPOSITION 2. *For any automaton (A, E^d, X, I) and for any $i, j \in E^d$, $i \overset{*}{\Rightarrow}_N j$ if and only if for any k, $1 \leqq k \leqq d$, there exist $m_1, \cdots, m_n, m_1', \cdots, m_n' \in Z^{+,0}$ such that $m_1\xi_1 + \cdots + m_n\xi_n = e_k$ and $m_1'\xi_1 + \cdots + m_n'\xi_n = -e_k$, where $Z^{+,0}$ denotes the nonnegative integers and the e_k are the standard basis elements.*

The following should be clear from the definitions.

PROPOSITION 3. *For any laminal subarrays $\Lambda_p, \Lambda_q \in E^d/R_N^*$, $p \neq q$ if and only if no $i \in \Lambda_p$ can be reached from any $j \in \Lambda_q$ by a forward* *neighborhood trace. Equivalently,* $(\forall\Lambda_p, \Lambda_q)((\Lambda_p = \Lambda_q) \Leftrightarrow (\exists i \in \Lambda_p)(\exists j \in \Lambda_q)(i \overset{*}{\Rightarrow}_N j))$.

PROPOSITION 4. *Cells i and j are in the same laminal subarray if and only if there exists a cell k that can be reached from both cells i and j by a forward trace. Also, cells i and j are in the same laminal subarray if and only if there exists a cell k that can reach both cells i and j by a forward trace.* I.e.,

$$(\forall i, j)((\exists\Lambda_p)(i, j \in \Lambda_p) \Leftrightarrow (\exists k)(i \overset{*}{\Rightarrow}_N k \;\&\; j \overset{*}{\Rightarrow}_N k)$$
$$\Leftrightarrow (\exists k)(k \overset{*}{\Rightarrow}_N i \;\&\; k \overset{*}{\Rightarrow}_N j)).$$

Proof. If iR_N^*j, then for at least one sequence $\xi_{t_1}, \xi_{t_2}, \cdots, \xi_{t_m}$, $i = j + k_1\xi_{t_1} + \cdots + k_m\xi_{t_m}$ where $k_s = 1$ or -1 and ξ_{t_s} in X, $1 \leqq s \leqq m$. Moving all terms with $k_s < 0$ to the left side, we get

$$i + \sum_{k_s<0} \xi_{t_s} = j + \sum_{k_s>0} \xi_{t_s} = k.$$

Cells i and j can therefore both reach cell k by forward traces.

Let

$$k' = i - \sum_{k_s>0} \xi_{t_s} = j - \sum_{k_s<0} \xi_{t_s},$$

then

$$i = k' + \sum_{k_s>0} \xi_{t_s},$$

$$j = k' + \sum_{k_s<0} \xi_{t_s}$$

and therefore cell k' can be reached from both cells i and j by forward traces. The converses are easily seen. ::

In the previous example, cells $(0, 0)$ and $(0, 4n)$, $n \in Z$, can trace forward to $(n, 2n)$. Note that they are all in Λ_0 and that $(0, 0)$ cannot be reached from $(0, 4n)$ by a forward trace, and vice versa.

THEOREM 5. *The rank of Λ_0 is d if and only if for each k, $1 \leqq k \leqq d$, there is a cell i in Λ_0 such that $i = ze_k$, $z \in Z$ and $z \neq 0$.*

Proof. (Left to right) Assume the rank of Λ_0 is d. Let the set of d generators of Λ_0 be $\Theta = \{\theta_i = (z_{i1}, \cdots, z_{id}) \mid z_{ij} \in Z, 1 \leqq i, j \leqq d\}$.

Consider the matrix

$$\bar{\bar{\Theta}} = \begin{Vmatrix} z_{11} & z_{12} & \cdots & z_{1d} \\ z_{21} & z_{22} & \cdots & z_{2d} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ z_{d1} & z_{d2} & \cdots & z_{dd} \end{Vmatrix}.$$

By replacing rows by nonzero multiples and by subtracting multiples of rows from appropriate multiples of other rows together with row permutations if necessary, we can easily arrive at another matrix of the form

$$\bar{\bar{\Theta}}' = \begin{Vmatrix} z_{11}' & 0 & 0 & \cdots & 0 & 0 \\ 0 & z_{22}' & 0 & \cdots & 0 & 0 \\ \cdot & \cdot & \cdot & & \cdot & \cdot \\ \cdot & \cdot & \cdot & & \cdot & \cdot \\ \cdot & \cdot & \cdot & & \cdot & \cdot \\ 0 & 0 & 0 & \cdots & 0 & z_{dd}' \end{Vmatrix},$$

where by construction each row is a d-tuple in Λ_0 of the required form. (Right to left) Assume the rank of Λ_0 is less than d. Since all $z_k e_k$ are mutually independent if $z_k \neq 0$, we see that not all such $z_k e_k$, $1 \leqq k \leqq d$, can be in Λ_0. ::

Note that unlike in a vector space, a rank of d for Λ_0 does not imply $\Lambda_0 = E^d$. Also, in general, the rows of $\bar{\bar{\Theta}}'$ are no longer a basis of Λ_0.

COROLLARY 5.1. *Let* $|\bar{\bar{\Theta}}|$ *be the determinant of the matrix* $\bar{\bar{\Theta}}$ *of generators of* Λ_0. *Then the d-tuples* $(|\bar{\bar{\Theta}}|, 0, \cdots, 0)$, $(0, |\bar{\bar{\Theta}}|, 0, \cdots, 0)$, $\cdots$, $(0, \cdots, 0, |\bar{\bar{\Theta}}|)$ *are all in* Λ_0.

Proof. $|\bar{\bar{\Theta}}| = |\bar{\bar{\Theta}}'|$, is the product of the diagonal terms in $\bar{\bar{\Theta}}'$. Therefore each d-tuple $|\bar{\bar{\Theta}}| e_i$ is a multiple of a row of $\bar{\bar{\Theta}}'$. ::

THEOREM 6. *The number of equivalence classes in* $E^d/R_N{}^*$ *is finite* *if and only if the rank of* Λ_0 *is equal to d, where d is the tessellation dimension.*

Proof. (Right to left) Assume the rank of Λ_0 is d. By Theorem 5 we know there are d independent elements of Λ_0 of the form $z_k e_k$. From Proposition 1 we know that any $\Lambda_j = \Lambda_0 + \zeta_j$, for some $\zeta_j \in Z^d$. With $\zeta_j = (\alpha_{j1}, \cdots, \alpha_{jd})$ we define *reduced* ζ_j by $\zeta_j^{rd} = (\beta_{j1}, \cdots, \beta_{jd})$ such that for each $k = 1, \cdots, d$, β_{jk} is the smallest positive integer such that $\beta_{jk} \equiv \alpha_{jk} \pmod{z_k}$. Since for any $k_i \in Z$, $1 \leqq i \leqq d$, $k_1(z_1 e_1) + \cdots + k_d(z_d e_d) \in \Lambda_0$, it is clear that $\Lambda_0 + \zeta_j = \Lambda_0 + \zeta_j^{rd}$. Since there are only finitely many reduced ζ_j, we can conclude $\#(E^d/R_N^*)$ is finite. (Left to right) Suppose the rank of Λ_0 is less than d. By Theorem 5 for at least one e_k there is no nonzero $z \in Z$ such that $z e_k \in \Lambda_0$. Hence for all $z_1, z_2 \in Z$, if $z_1 \neq z_2$ then $\Lambda_0 + z_1 e_k \neq \Lambda_0 + z_2 e_k$, otherwise $(z_1 - z_2) e_k \in \Lambda_0$. ::

With X_{-0} denoting the set of all non-0^d components of X, we have

COROLLARY 6.1. *For any automaton* $M = (A, E^d, X, I)$, *if* $\#(X_{-0}) < d$, *then M is infinitely laminated.*

PROPOSITION 7. *For any positive integer z, there exists an automaton* (A, E^d, X, I) *with a neighborhood index X with d or more components, such that* $\#(E^d/R_N^*) = z$.

Proof. Take the first $d - 1$ components of X to be unit vectors along $d - 1$ coordinate axes. Take the dth component of X to be a length z vector along the last coordinate axis. These d components define the desired lamination. The remaining components of X, if any, are taken in Λ_0. ::

THEOREM 8. *For any submodule* Λ_0 *of* E^d, *there is an automaton* (A, E^d, X, I) *such that* $\Lambda_0 \in E^d/R_N^*$, *and X is reduced.*

Proof. If $\{\theta_1, \cdots, \theta_r\}$ is a basis of Λ_0, choose $X = (\theta_1, \cdots, \theta_r)$ and by choosing $I = T$, we ensure that X is reduced. ::

PROPOSITION 9. *For any positive integer k, there are infinitely many X such that* $\#(X) = k$ *and* $\#(E^d/\Lambda_0(X)) = \aleph_0$.

Proof. Choose the components of X from the elements of a nontrivial submodule Λ_0' of E^d of rank less than d. Then $\Lambda_0(X) \subseteq \Lambda_0'$. By Corollary 6.1, $\#(E^d/\Lambda_0') = \aleph_0$; it follows that $\#(E^d/\Lambda_0(X)) = \aleph_0$ also. ::

With $\#(X_{-0})$ denoting the number of non-0^d components of X, we now show

PROPOSITION 10. *If* $\Lambda_0(X)$ *is a nontrivial submodule of* E^d *where* $\#(X_{-0}) \geqq 2$, *then there are infinitely many* X_i *such that* $\Lambda_0(X) = \Lambda_0(X_i)$.

Proof. If $\Lambda_0(X)$ has rank 1, then one component of X can be the generator of $\Lambda_0(X)$ and a second component can be an arbitrary non-0^d element of $\Lambda_0(X)$. If $\Lambda_0(X)$ has rank greater than one, consider any pair of elements θ_i and θ_j of a basis Θ of $\Lambda_0(X)$. Replace θ_i by $\theta_k = 2\theta_j - \theta_i$ and denote the resulting set by Θ'. Since $\theta_i = 2\theta_j - \theta_k$, Θ' is also a basis of $\Lambda_0(X)$. We can repeat this process on Θ', now treating the pair θ_j and θ_k as we did θ_i and θ_j for Θ, i.e., we replace θ_j by $2\theta_k - \theta_j$ obtaining another Θ''. This process can be repeated indefinitely, each time giving a new basis which when used as a neighborhood index, generates the same laminal submodule. ::

COROLLARY 10.1. *There are a denumerably infinite number of bases of* E^d.

PROPOSITION 11. *For any positive integers* k, z *with* $k \geqq d$, *there are infinitely many* X *such that* $\#(X_{-0}) = k$ *and* $\#(E^d/\Lambda_0(X)) = z$.

Proof. From Theorem 6 and Proposition 7 we know there is an X with $\#(X_{-0}) = d$ that will generate a $\Lambda_0(X)$ such that for any positive integer z, $\#(E^d/\Lambda_0(X)) = z$. If $k > d$, the remaining components can be arbitrarily chosen from $\Lambda_0(X)$. That infinitely many such X exist follows from Proposition 10. ::

As we have seen, if an automaton is laminated, cells in different laminal subarrays cannot interchange information. Such an automaton $M = (A, E^d, X, I)$ can therefore be thought of as consisting of the partial automata (A, Λ_0, X, I), (A, Λ_1, X, I), $\cdots$, acting in parallel, where $E^d/R_N{}^* = \{\Lambda_0, \Lambda_1, \cdots\}$. Each of these partial automata will be called a *laminal subautomaton*, and will be denoted by $M(\Lambda_k)$.

IV. HOMOMORPHISMS AND LAMINAL SUBAUTOMATA

In the preceding section we saw that an automaton M is laminated if and only if X does not contain a basis of Z^d among its components. In this section we show by means of certain homomorphisms that each laminal subautomaton is isomorphic to an automaton.

It is possible to define various homomorphisms among automata and laminal subautomata. We now examine two such homomorphisms. Consider $M^{(1)} = (A^{(1)}, E^{(1)}, X^{(1)}, I^{(1)})$ with lamination $\{\Lambda_0^{(1)}, \Lambda_1^{(1)}, \cdots\}$ and $M^{(2)} = (A^{(2)}, E^{(2)}, X^{(2)}, I^{(2)})$ with lamination $\{\Lambda_0^{(2)}, \Lambda_1^{(2)}, \cdots\}$ where $\#(X^{(1)}) = \#(X^{(2)}) = n$. (Note that the earlier convention of superscripts to E's indicating tessellation dimension is altered here for convenience. Also $\#$ is extended to apply to n-tuples to indicate the number of components.) A quadruple of mappings $\mu_s = (\mu_a, \mu_e, \mu_x, \mu_\sigma)$ is said to be a *structural homomorphism* from $M^{(1)}(\Lambda_p^{(1)})$ into $M^{(2)}(\Lambda_q^{(2)})$, denoted by $M^{(1)}(\Lambda_p^{(1)}) \xrightarrow{\sim}_{\mu_s} M^{(2)}(\Lambda_q^{(2)})$ if and only if

$\mu_a: A^{(1)} \to A^{(2)}$,

$\mu_e: \Lambda_p^{(1)} \to \Lambda_q^{(2)}$,

$\mu_x: \underline{X}^{(1)} \to \underline{X}^{(2)}$ and is one-to-one and onto, where $\underline{X}^{(i)}$ is the set of the components of $X^{(i)}$, $i = 1, 2$,

$\mu_\sigma: J^{(1)} \to J^{(2)}$, where $J^{(1)}$ and $J^{(2)}$ are the sets of local transformations defining $I^{(1)}$ and $I^{(2)}$, respectively (i.e., $J^{(1)} = \delta_{X^{(1)}}^{-1}(I^{(1)})$

$$\begin{aligned}&\text{and } J^{(2)} = \delta_{X^{(2)}}^{-1}(I^{(2)})\text{), such that } (\forall k,\ 1 \leqq k \leqq d)(\forall i \in \Lambda_p)\\ &(\mu_e[(N(X^{(1)}, i))]_k = [N(\mu_x(X^{(1)}), \mu_e(i))]_k)\end{aligned} \tag{1}$$

where subscript k denotes the kth component of the n-tuple, and

$$\begin{aligned}(\forall(a_{i_1}, \cdots, a_{i_n}) \in (A^{(1)})^n)(\forall\sigma_j^{(1)} \in J^{(1)})(\mu_a(\sigma_j^{(1)}(a_{i_1}, \cdots, a_{i_n}))\\ = (\mu_\sigma\sigma_j^{(1)})[\mu_a(a_{i_1}), \cdots, \mu_a(a_{i_n})]).\end{aligned} \tag{2}$$

A *structural isomorphism* between $M^{(1)}(\Lambda_p^{(1)})$ and $M^{(2)}(\Lambda_q^{(2)})$ is defined by $M^{(1)}(\Lambda_p^{(1)}) \overset{\sim}{\leftrightarrow}_{\mu_s} M^{(2)}(\Lambda_q^{(2)}) \Leftrightarrow M^{(1)}(\Lambda_p^{(1)}) \xrightarrow{\sim}_{\mu_s} M^{(2)}(\Lambda_q^{(2)})$ & $M^{(2)}(\Lambda_q^{(2)}) \xrightarrow{\sim}_{\mu_s^{-1}} M^{(1)}(\Lambda_p^{(1)})$ where $\mu_s^{-1} = (\mu_a^{-1}, \mu_e^{-1}, \mu_x^{-1}, \mu_\sigma^{-1})$.

For any laminated automaton, it is immediate that, if μ_e is the mapping defined by translation by some $\zeta \in Z^d$ such that $\Lambda_p = \Lambda_q + \zeta$, and if μ_a, μ_x, μ_σ are identity mappings, then $M(\Lambda_p) \overset{\sim}{\leftrightarrow}_{\mu_s} M(\Lambda_q)$.

Given a configuration c and a lamination Λ, by $c(\Lambda)$ we mean the restriction of c to the cells in Λ. We call $c(\Lambda)$ the *subconfiguration induced by* Λ. If $C(\Lambda) = \{c(\Lambda) \mid c \in C\}$ then a laminal subautomaton can be expressed by $M(\Lambda) = (C(\Lambda), I, \psi)$.

Given two automata $M^{(1)} = (C^{(1)}, I^{(1)}, \psi^{(1)})$ with lamination $\{\Lambda_0^{(1)}, \Lambda_1^{(1)}, \cdots\}$ and $M^{(2)} = (C^{(2)}, I^{(2)}, \psi^{(2)})$ with lamination $\{\Lambda_0^{(2)}, \Lambda_1^{(2)}, \cdots\}$, an ordered pair of mappings $\mu_b = (\mu_c, \mu_\tau)$ is said to be a *behavioral homomorphism* from $M^{(1)}(\Lambda_p^{(1)})$ into $M^{(2)}(\Lambda_q^{(2)})$ denoted by $M^{(1)}(\Lambda_p^{(1)}) \xrightarrow{\sim}_{\mu_b} M^{(2)}(\Lambda_q^{(2)})$ if and only if

$\mu_c: C^{(1)}(\Lambda_p^{(1)}) \to C^{(2)}(\Lambda_q^{(2)})$, and

$\mu_\tau: I^{(1)} \to I^{(2)}$ such that

$$(\forall c \in C^{(1)}(\Lambda_p^{(1)}))(\forall \tau \in I^{(1)})(\mu_c(\psi^{(1)}(c, \tau)) = \psi^{(2)}(\mu_c(c), \mu_\tau(\tau))). \quad (3)$$

A *behavioral isomorphism* between $M^{(1)}(\Lambda_p^{(1)})$ and $M_2(\Lambda_q^{(2)})$ is defined by $M^{(1)}(\Lambda_p^{(1)}) \overset{\sim}{\leftrightarrow}_{\mu_b} M^{(2)}(\Lambda_q^{(2)}) \Leftrightarrow M^{(1)}(\Lambda_p^{(1)}) \xrightarrow{\sim}_{\mu_b} M^{(2)}(\Lambda_q^{(2)})$ & $M^{(2)}(\Lambda_q^{(2)}) \xrightarrow{\sim}_{\mu_b^{-1}} M^{(1)}(\Lambda_p^{(1)})$ where $\mu_b^{-1} = (\mu_c^{-1}, \mu_b^{-1})$.

THEOREM 1. *Let $M^{(1)}(\Lambda_p^{(1)})$ and $M^{(2)}(\Lambda_q^{(2)})$ be laminal subautomata of arbitrary automata $M^{(1)}$ and $M^{(2)}$, respectively. If there exists a structural homomorphism from $M^{(1)}(\Lambda_p^{(1)})$ into $M^{(2)}(\Lambda_q^{(2)})$, then there exists a behavioral homomorphism from $M^{(1)}(\Lambda_p^{(1)})$ into $M^{(2)}(\Lambda_q^{(2)})$.*

Proof. Let $M^{(1)} = (A^{(1)}, E^{(1)}, X^{(1)}, I^{(1)}) = (C^{(1)}, I^{(1)}, \psi^{(1)})$, $M^{(2)} = (A^{(2)}, E^{(2)}, X^{(2)}, I^{(2)}) = (C^{(2)}, I^{(2)}, \psi^{(2)})$, and let $\mu_s = (\mu_a, \mu_e, \mu_x, \mu_\sigma)$ be a structural homomorphism from $M^{(1)}(\Lambda_p^{(1)})$ to $M^{(2)}(\Lambda_q^{(2)})$.

We define $\mu_c: C^{(1)}(\Lambda_p^{(1)}) \to C^{(2)}(\Lambda_q^{(2)})$ by

$$\begin{aligned}&(\forall c^{(1)} \in C^{(1)}(\Lambda_p^{(1)}))(\forall c^{(2)} \in C^{(2)}(\Lambda_q^{(2)}))\\ &\quad\cdot[\mu_c(c^{(1)}) = c^{(2)} \Leftrightarrow (\forall i \in \Lambda_p^{(1)})(\mu_a(c^{(1)}(i)) = c^{(2)}(\mu_e(i)))].\end{aligned} \quad (4)$$

We define $\mu_\tau: I^{(1)} \to I^{(2)}$ as follows. For any $\tau^{(1)} \in I^{(1)}$, let $\sigma^{(1)} = (\delta^{(1)})^{-1}(\tau^{(1)}, X^{(1)})$, where the inverse of $\delta^{(1)}$ is taken with $X^{(1)}$ fixed, then

$$(\forall \sigma^{(1)} \in L^{(1)})[\mu_\tau(\delta^{(1)}(\sigma^{(1)}, X^{(1)})) = \delta^{(2)}(\mu_\sigma(\sigma^{(1)}), \mu_x(X^{(1)}))]. \quad (5)$$

We now show that $\mu_b = (\mu_c, \mu_\tau)$ is the required behavioral homomorphism by establishing the following.

For $c_j^{(1)} \in C^{(1)}(\Lambda_p^{(1)})$ and $\tau^{(1)} \in I^{(1)}$, if $\tau^{(1)}(c_j^{(1)}) = c_k^{(1)}$,

then $$(\mu_\tau\tau^{(1)})[\mu_c(c_j^{(1)})] = \mu_c(c_k^{(1)}).$$

Assume $\tau^{(1)}(c_j^{(1)}) = c_k^{(1)}$. By (5), we have

$$\begin{aligned}(\mu_\tau\tau^{(1)})[\mu_c(c_j^{(1)})] &= (\mu_\tau\delta^{(1)}(\sigma^{(1)}, X^{(1)}))[\mu_c(c_j^{(1)})]\\ &= \delta^{(2)}(\mu_\sigma(\sigma^{(1)}), \mu_x(X^{(1)}))[\mu_c(c_j^{(1)})].\end{aligned}$$

The $\mu_e(i)$th cell of this latter configuration contains $(\mu_\sigma\sigma^{(1)})$.

$[\mu_c(c_j^{(1)})(N(\mu_x(X^{(1)}), \mu_e(i)))]$ since, in general, $(\tau(c))(i) = \sigma(c(N(X, i)))$. By (1), the above term is equal to

$$\begin{aligned}(\mu_\sigma\sigma^{(1)})[\mu_c(c_j^{(1)})[\mu_e(N(X^{(1)}, i))]] &= (\mu_\sigma\sigma^{(1)})[\mu_a(c_j^{(1)}(N(X^{(1)}, i)))] && \text{by (4),}\\ &= \mu_a(\sigma^{(1)}(c_j^{(1)}(N(X^{(1)}, i)))) && \text{by (2),}\\ &= \mu_a(\sigma^{(1)}(c_j^{(1)})(i))\\ &= \mu_a(c_k^{(1)}(i)).\end{aligned}$$

Since i was arbitrary over $\Lambda_p^{(1)}$, we conclude

$$(\mu_\tau\tau^{(1)})(\mu_c(c_j^{(1)})) = \mu_c(c_k^{(1)})$$

and that μ_b is a behavioral homomorphism. ::

Just as in the example of Section III, if $\Lambda_q = \Lambda_p + \zeta$, then $C(\Lambda_q) = C(\Lambda_p + \zeta)$. $M(\Lambda_p)$ and $M(\Lambda_q)$ will be behaviorally isomorphic, and it will therefore be sufficient to focus attention only on $M(\Lambda_0)$. Since Λ_0 is a submodule of Z^d, it will have a basis $\Theta = (\theta_1, \cdots, \theta_r)$, where $r \leqq d$. The rank of Λ_0 will also be referred to as the rank of the laminal subautomaton $M(\Lambda_0)$.

By letting $\Lambda_0^{(1)} = E^{(1)}$ and $\Lambda_0^{(2)} = E^{(2)}$, we obtain

COROLLARY 1.1. *If there exists a structural homomorphism from $M^{(1)}$ onto $M^{(2)}$, then there exists a behavioral homomorphism from $M^{(1)}$ onto $M^{(2)}$.*

The converses of Theorem 1 and Corollary 1.1 are not true. These results naturally fit in the development reported in Yamada–Amoroso (1968b).

THEOREM 2. *For a given laminal subautomaton $M^{(1)}(\Lambda_0^{(1)})$ of rank r of $M^{(1)} = (A, E^d, X^{(1)}, I^{(1)})$, there exists a nonlaminal subautomaton $M^{(2)} = (A, E^r, X^{(2)}, I^{(2)})$ (of dimension r) such that $M^{(1)}(\Lambda_0^{(1)})$ is structurally and behaviorally isomorphic to $M^{(2)}$.*

Proof. With $\Theta = (\theta_1, \cdots, \theta_r)$ a basis of $\Lambda_0^{(1)}$, and $X^{(1)} = (\xi_1^{(1)}, \cdots, \xi_r^{(1)})$, define $\mu_s = (\mu_a, \mu_e, \mu_x, \mu_\sigma)$ as follows. μ_a and μ_σ are identity mappings. For any $k_1\theta_1 + \cdots + k_r\theta_r \in \Lambda_0$, $\mu_e(k_1\theta_1 + \cdots + k_r\theta_r) = (k_1, \cdots, k_r) \in E^r$. Clearly, the range of μ_e is E^r. Finally if $\xi_j^{(1)} = \alpha_{j1}\theta_1 + \cdots + \alpha_{jr}\theta_r$, then $\mu_x(\xi_j^{(1)}) = (\alpha_{j1}, \cdots, \alpha_{jr})$. It is easy to verify that the range of μ_x is a basis for E^r, showing $M^{(2)}$ is nonlaminal, and it is easy to verify that μ_s is a structural isomorphism. The existence of a behavioral isomorphism follows from the last theorem. ::

Theorem 2 serves as a justification for our limiting ourselves henceforth to only nonlaminal automata. We shall call a nonlaminal automaton $M^{(2)}$, isomorphic to a laminal subautomaton $M^{(1)}(\Lambda_0)$, a *lamination reduced* automaton of $M^{(1)}$. We shall refer to this reduction procedure as the *lamination reduction* of an automaton. A neighborhood index which gives a laminated automaton will be referred to as a *laminating index* (otherwise, a nonlaminating index).

With respect to a fixed E^d, let $\Lambda_0(X)$ denote the laminal submodule of E^d defined by the neighborhood index X. If we partition $\Xi \nsubseteq \bigcup_{n=0}^{\infty} (Z^d)^n$, the set of all possible neighborhood indices, by the equivalence relation R_Λ, defined by

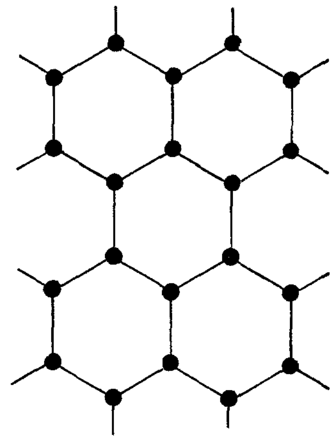
$$(\forall X_i, X_j \in \Xi)(X_i R_\Lambda X_j \Leftrightarrow \Lambda_0(X_i) = \Lambda_0(X_j)),$$

then these classes would form a lattice equal to the lattice of all submodules of Z^d.

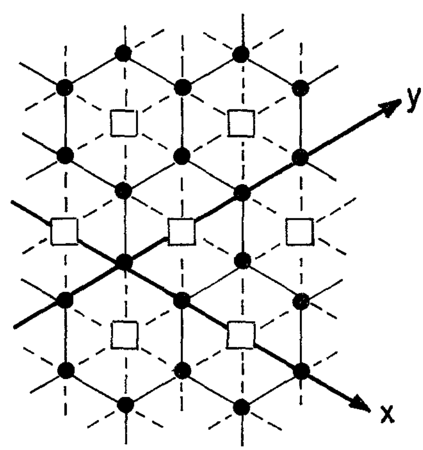
In summary we note that any automaton M with $\#(X_{-0}) < d$ is necessarily laminated, and that there exists an automaton M' in dimension $d' \leqq \#(X_{-0})$ such that M' is structurally isomorphic to the laminal subautomata of M.

V. SOME GENERALIZATIONS OF TESSELLATION AUTOMATA

We defined the cells of a tessellation automaton in terms of the lattice points in Euclidean space. Alternatively, the cells could have been associated with some other tessellation, e.g., with the vertices of a hexagonal tessellation as shown in Fig. 1 (a), where the neighborhood of any cell is the cell itself and the three closest cells. Although this array seems to be essentially different, it turns out that any regular polytope array can be embedded in the lattice point tessellations. We shall not prove this here. Rather we shall only indicate how the above example can be handled. If we use the oblique coordinate system shown in Fig. 1 (b) and make all the cells at coordinate points $\{(3m + 2 + n, n) \mid m, n \in Z\}$ "dead" cells, then by letting $X = ((0, 0), (-1, 0), (-1, 1), (0, 1), (1, 0), (1, -1), (0, -1))$, we have the hexagonal tessellation embedded in the lattice point tessellation. The dead cells are handled as follows. State alphabet A is enlarged by adding a new state D which is assumed only by the dead cells. For each local transformation σ_j, $\sigma_j(y_1, \cdots, y_7) = D$ if and only if $y_1 = D$. If $y_1 \neq D$ and $y_3 = D$, then $\sigma_j(y_1, \cdots, y_7)$ is determined by y_1, y_2, y_4, and y_6 only. If $y_1 \neq D$ and $y_6 = D$, then $\sigma_j(y_1, \cdots, y_7)$ is determined by y_1, y_3, y_5, and y_7



(a) HEXAGONAL TESSELLATION



(b) EMBEDDING IN SQUARE TESSELLATION

FIG. 1. Embedding of hexagonal tessellation

only. Within these limitations, the σ_j are defined to simulate the hexagonal automaton.

With A and I appropriately constructed, the tessellation automaton is the natural structure for the relaxation method for the solution of linear partial differential equations. The initial conditions are obtained by properly selecting the initial configuration, and boundary conditions can be handled with the help of the dead cell concept discussed above.

If we allow A and X to be infinite sets of cardinality $\aleph_1$, then such tessellation automata can be identified with continuous physical systems, the local transformation being an integration. Newtonian mechanics can be embedded in such a system, and hence this generalization could give rise to approximation models for many physical systems.

It can easily be shown that with fixed and finite neighborhood indices, the successive application of two transformations in I is usually not equivalent to the application of some single transformation in I. If we allow neighborhood indices to be infinite sequences, then nontrivial input alphabets I can be made to be semigroups under the operation of successive application. These semigroups seem worthy of further investigation.

Returning now to the automata studied in this paper, note that for any given automaton, the neighborhood index was fixed and the local transformation varied to account for the configuration changes. Alternatively, fixing $\#(X)$ and some local transformation, the neighborhood indices could be varied. I.e., we could define automata $M_{jk} = (A, E^{\mathrm{d}}, G_j, \sigma_k)$ where $G_j \subseteq \Xi^{(d,n)}$, $\sigma_k \in L^{(A,n)}$, or behaviorally by $M_{jk} = (C, S^{(A,d,n)}, \psi)$ where $S_k^{(A,d,n)} = \{\delta(\sigma_k^{(A,n)}, X_j^{(d,n)}) \mid X_j^{(d,n)} \in \Xi^{(d,n)}\}$ and $\psi : C \times S_k^{(A,d,n)} \to C$.

Other classes of automata can be defined by allowing variations in both neighborhood index and local transformation.

Finally, we have been requiring that the same local transformation apply throughout the array at any given time. We can, however, allow regional differences which perhaps could vary according to certain rules. It appears that in order to accommodate the analysis of some of the cellular logic arrays discussed in Minnick (1967), such a generalization might be required. We have in mind the locally varied external input signals. A further generalization along these lines might also be applicable to the relaxation method solution of nonlinear partial differential equations.

VI. CONCLUDING REMARKS

This report is the first of a series in which we hope to develop a mathematical framework (the tessellation automaton) in which one can rigorously study the wide variety of existing systems whose underlying structure is that of a uniformly interconnected array of indentically constructed logical devices. More specifically, in this initial report we have introduced the general d-dimensional tessellation automaton together with certain properties (e.g., the laminal structure, the structural and behavioral homomorphisms) which will play a central role in the structural theory of tessellation automata that will be outlined in Yamada–Amoroso (1968a), (1968b), and (1968c). This theory will be concerned with such concepts as structural reductions preserving certain processing capabilities, and with classes of tessellation automata equivalent in certain meaningful senses and representations of these classes. Certain behavioral questions are treated in Yamada–Amoroso (1968d) and (1968e).

As this work has progressed we have formulated a great number of questions most of which are as yet untouched. Therefore, this report and even the ones to follow should not be considered a detailed treatment of these ideas, but rather as an indication of some aspects of a broad topic that others may wish to help develop.

We are certain that many already established mathematical results (especially from algebra and number theory) can easily be applied to further uncover properties of tessellation automata. We are equally certain, however, that applications and engineering motivations will require that the theory of tessellation automata ultimately develop independently. In fact, it appears that some new mathematical concepts will be required before a complete understanding of these structures comes about.

Received: September 20, 1968; revised January 29, 1969

REFERENCES

Barricelli, N. A. (1957), Symbiogenetic Evolution Processes Realized by Artificial Methods, *Methodos* **9**, 148–182.

Barricelli, N. A. (1963), Numerical Testing of Evolution Theories, *Acta Biotheoretica* **16**, 69–126.

Cohn, H. (1962), "A Second Course in Number Theory," Wiley, New York.

Harrison, M. A. (1965), "Introduction to Switching and Automata Theory," McGraw-Hill, New York.

JACOBSON, N. (1953), "Lectures in Abstract Algebra," Vol. 2, Van Nostrand, New York.

LEE, C. Y. (1962), Intercommunicating Cells, Basis for a Distributed Logic Computer, *Proc. AFIPS Fall Joint Computer Conference* **22**, 130–136.

LEE, C. Y. (1963), Synthesis of a Cellular Computer Using the 29-State Model of von Neumann, *Bell Laboratories Memorandum*, MM63-344-5.

LEE, C. Y., AND PAULL, M. C. (1963), A Content Addressable Distributed Logic Memory with Applications to Information Retrieval, *Proc. I.R.E.* **51**, 924–932.

MINNICK, R. C. (1967), A Survey of Microcellular Research, *J. A.C.M.* **14**, 203–241.

MOORE, E. F. (1962), Machine Models of Self-Reproduction, *Proc. Symp. Appl. Math.* **14**, 17–33.

MYHILL, J. (1963), A Converse to Moore's Garden-of-Eden Theorem, *Proc. Am. Math. Soc.* **14**, 685–686.

SLOTNICK, D. L., BORCK, W. C. AND MCREYNOLDS, R. C. (1962), The Solomon Computer, *Proc. AFIPS Fall Joint Computer Conference* **22**, 97–107.

THATCHER, J. W. (1964), "Universality in the von Neumann Cellular Model," I.B.M. Research Report RC-1255, 100 pp.

UNGER, S. H. (1958), A Computer Oriented Toward Spatial Problems, *Proc. I.R.E.* **46**, 1744–1750.

UNGER, S. H. (1959), Pattern Detection and Recognition, *Proc. I.R.E.* **47**, 1732–1752.

VON NEUMANN, J. (1966), "Theory of Self-Reproducing Automata," Univ. of Illinois Press, Urbana.

YAMADA, H. AND AMOROSO, S., (1968a), Certain Equivalences and Representations of Tessellation Automata, (submitted for publication).

YAMADA, H., AND AMOROSO, S., (1968b), Blocking and Equivalences of Tessellation Automata, (in preparation).

YAMADA, H., AND AMOROSO, S., (1968c), Tessellation Automata with Initial Configurations, (in preparation).

YAMADA, H., AND AMOROSO, S. (1968d). A Completeness Problem for Tessellation Automata, (submitted for publication).

YAMADA, H. AND AMOROSO, S. (1968e), Pattern Decompositions in Tessellation Automata with an Application to the Completeness Problem, (submitted for publication).