Integral Sequential Word Functions and Growth Equivalence of Lindenmayer Systems

Azaria Paz

Computer Science Department, Technion, Haifa, Israel

And

Arto Salomaa

Mathematics Department, University of Turku, Finland

Growth functions of informationless Lindenmayer systems are investigated from the point of view of integral sequential word functions. Algorithms are obtained for the solution of equivalence, minimization and construction problems. It is found out that some of the inclusion relations between language families do not remain valid for the corresponding families of growth functions. Some results concerning context-dependent Lindenmayer systems, as well as growth relations of OL-systems are also obtained.

1. Introduction

Lindenmayer systems (also called L-systems, Lindenmayer models or developmental languages) have been the object of extensive study during the past two years. The systems were introduced in connection with a theory proposed to model the development of filamentous organisms. The stages of development are represented by words corresponding to one-dimensional arrays of cells (filaments). The developmental instructions are modelled by ordinary rewriting rules or productions. These productions are applied simultaneously to all letters to reflect the simultaneity of the growth in the organism. This parallel rewriting is the main difference between Lindenmayer systems and ordinary generative grammars. There are many types of Lindenmayer systems. One distinction results from the fact that the various parts of the developing organism may or may not be in communication with each other.
other. Different types of systems will be defined in the sequel at appropriate places. For more background material and motivation, the reader is referred to Rozenberg and Doucet (1971), Rozenberg and Lindenmayer (1971) and Salomaa (1973), as well as to the items given in their bibliographies.

A particularly interesting aspect in the study of Lindenmayer systems is the theory of growth functions. The basic paper in this field is by Szilard (1972). In the theory of growth functions only the lengths of the words matter, no attention is paid to the words themselves. This implies that many problems become solvable whose solution is unknown for L-systems in general. Also hierarchies of language families may reduce to one family of growth functions.

The basic observation behind this paper is that growth functions of certain Lindenmayer systems fit in the framework of the theory of integral sequential word functions. Functions resembling the latter have been studied extensively in the past, cf. Paz (1971, pp. 116-144) in connection with probabilistic automata. Consequently, our subsequent results might be of interest to both people working with word functions and to people interested in Lindenmayer systems. The former may read only Section 2 of this paper, although some definitions get their motivation in Section 3. (One of them is the definition of the vector \( \eta \).) On the other hand, people interested in Lindenmayer systems may read Section 3 only, although they then miss many of the proofs.

Basic notions concerning integral sequential word functions are introduced in Sections 2a and 2b. Some theorems, based on earlier results, are also mentioned. Section 2c gives preliminary results concerning the reduction problem which is then attacked in its general form in Section 2d. The main result is the general reduction Theorem 13 which gives a solution to the minimization problem and is directly applicable to growth functions. Using some results previously known, the reduction theorem is then applied to solve the problem of realizing a given function as an integral sequential word function (Theorem 17). Some other representability problems are also considered in Section 2e. Section 2f deals with closure properties, and Section 2g with the single letter case which, in fact, corresponds to DOL-systems.

Section 3a deals with the growth functions of DOL-systems. Algorithms are given for the solution of the following problems: growth equivalence, finding all growth equivalent axioms and cell minimization. It is also shown that, for any DOL-system \( S \) and integer \( k \), there is only a finite number of DOL-systems growth equivalent to \( S \) and having \( k \) letters in their alphabet. We also study the problem of realizing a given function as a growth function, as well as problems concerning malignant growth. The following Section 3b
deals with growth functions of context-dependent Lindenmayer systems. Examples are given of such growth functions which are not DOL growth functions. Also a result concerning the "saving of cells" in the transition from informationless to context-dependent systems is established. The last Section 3c deals with OL-systems and deterministic Lindenmayer systems with tables. For obvious reasons, the growth functions in these cases become growth relations. It is shown that the family of growth relations of DTOL-systems properly includes the family of growth relations of OL-systems, although mutual overlap holds between the corresponding language families.

2. Integral Sequential Word Functions

In this section we shall study integral sequential word functions; i.e., functions \( f: \Sigma^* \rightarrow N \) (\( \Sigma^* \) is the set of all words over a finite alphabet and \( N \) is the set of nonnegative integers) induced by a sequential integral system. The specific functions to be considered here can be used for investigating growth functions of various types of OL-systems as explained before. On the other hand, similar functions of a more general character have been studied elsewhere (see Paz (1971)) so that many theorems valid in the general case, carry over to this specific model. Whenever a proof to a theorem stated here is similar to an existing proof in the literature, we shall skip the proof here and refer the interested reader to the literature. We shall discuss here, in detail, only those aspects of the integral word functions which are pertinent to their use as a growth function and which exhibit a specific aspect different from the general case and resulting from the specific integral assumption.

a. Definitions and Notations

All the vectors and matrices considered in this section are assumed to have only nonnegative integral entries unless otherwise stated. A state vector is a vector having exactly one nonzero value. The notation \( \gamma \) stands for a column vector of due dimension with all its entries equal to 1. The notation \( \pi \) will be used for row vectors. Superscripts for vectors will be used for distinguishing between them and subscripts will be used to denote a specific entry in a vector. \( \Sigma \) denotes a finite alphabet, \( \Sigma^* \) the set of all words over \( \Sigma \), \( \lambda \), the empty word, and \( \sigma \), an element of \( \Sigma \).

DEFINITION 1. An \( n \)-state integral sequential system (IS) over a finite alphabet \( \Sigma \) is a triple \( \mathcal{A}_n = (\pi, \{ A(\sigma) \}_{\sigma \in \Sigma}, \gamma) \) where \( \pi \) is an \( n \)-dimensional "initial" row vector and the \( A(\sigma) \) are \( n \)-dimensional matrices. When using
the notation $\mathcal{A}$ instead of $\mathcal{A}_n$, we shall assume that the initial vector $\pi$ is not yet specified, while $\mathcal{A}_{n1}$ and $\mathcal{A}_{n2}$ will denote two IS which differ only in their initial vector $\pi$.

**DEFINITION 2.** An integral sequential word function (ISF) induced by the IS $\mathcal{A}$ is a function $f: \Sigma^* \to N$ (the superscript will be omitted when context is clear) defined as $f(x) = \pi A(x) \pi^2$ where $x = \sigma_1 \cdots \sigma_k \in \Sigma^*$ and $A(x) = A(\sigma_1) \cdots A(\sigma_k)$ by definition. Also by definition, $f(\lambda) = \pi \eta$.

**DEFINITION 3.** Two initial vectors $\pi^1$ and $\pi^2$ for a given IS $\mathcal{A}$ are equivalent if $f_{\mathcal{A}}(\pi^1(x)) = f_{\mathcal{A}}(\pi^2(x))$ for all $x \in \Sigma^*$.

**DEFINITION 4.** Two IS $\mathcal{A}_{n1}$ and $\mathcal{A}_{n2}$ over the same alphabet $\Sigma$, are equivalent if

$$f_{\mathcal{A}_{n1}}(x) = f_{\mathcal{A}_{n2}}(x)$$

for all $x \in \Sigma^*$.

**DEFINITION 5.** Two IS $\mathcal{A}^1$ and $\mathcal{A}^2$ are state equivalent if for any initial state vector $\pi^1$ for the first IS one can find an initial state vector $\pi^2$ for the second such that $\mathcal{A}^1_{\pi^1}$ is equivalent to $\mathcal{A}^2_{\pi^2}$ (notation: $\mathcal{A}^1_{\pi^1} \cong \mathcal{A}^2_{\pi^2}$), and vice-versa.

**Remark.** One verifies easily that if $\mathcal{A}^1$ is state equivalent to $\mathcal{A}^2$ then for any initial vector $\pi^1$ (not necessarily a state vector) there is an initial vector $\pi^2$ for $\mathcal{A}^2$ such that $\mathcal{A}^1_{\pi^1} \cong \mathcal{A}^2_{\pi^2}$ and vice-versa.

Given an IS $\mathcal{A}$, $K^\mathcal{A}$ and $G^\mathcal{A}$ denote the ordered infinite sets of column and row vectors, respectively:

$$K^\mathcal{A} = [\eta(\lambda), \eta(x_1), \ldots, \eta(x_k), \ldots]; \quad G^\mathcal{A} = \begin{bmatrix} \pi(\lambda) \\
\pi(x_1) \\
\pi(x_2) \\
\vdots \\
\pi(x_k) \end{bmatrix}$$

where by definition $\eta(x) = A(x) \eta$; $\eta(\lambda) = \eta$; $\pi(x) = \pi A(x)$; $\pi(\lambda) = \pi$; and $x_1 x_2 \cdots$ is a fixed lexicographic order on the words in $\Sigma^*$.

Let $K(m)$ and $G(m)$ denote the ordered subsets of $K^\mathcal{A}$ and $G^\mathcal{A}$, respectively, such that $\eta(x) \in K^\mathcal{A}(m) (\pi(x) \in G^\mathcal{A})$ for all $x$ such that $|x| \leq m$ ($|x|$ denotes the length of $x$).
b. Some Basic Theorems

THEOREM 1. For a given \( n \)-state IS \( \mathcal{A} \), there exists an effective algorithm for finding a set of linearly independent vectors in \( K^\mathcal{A}(n - 1) \) such that all vectors in \( K^\mathcal{A} \) depend linearly on them, and the same is true for \( G^\mathcal{A}_n(n - 1) \).

For proof see Paz (1971, p. 19).

Let \( \eta^1 \cdots \eta^m \) and \( \pi^1 \cdots \pi^k \) be the two sets of vectors having the following properties:

1. \( \eta^1 = \eta; \pi^1 = \pi \)
2. \( \eta^1, \ldots, \eta^m \) and \( \pi^1, \ldots, \pi^k \) are the first vectors in \( K^\mathcal{A} \) and \( G^\mathcal{A}_n \), respectively, according to the preassigned fixed order—which are linearly independent and span their whole sets.

The matrices \( H^\mathcal{A} \) and \( L^\mathcal{A}_n \) are defined as

\[
H^\mathcal{A} = [\eta^1 \cdots \eta^m], \quad L^\mathcal{A}_n = \begin{bmatrix} \pi^1 \\ \vdots \\ \pi^k \end{bmatrix}.
\]

It is clear that the ranks of the above defined matrices are \( \leq n \).

THEOREM 2. Two initial vectors for a given IS \( \mathcal{A} \) are equivalent if and only if

\[
\pi^1 H^\mathcal{A} = \pi^2 H^\mathcal{A}.
\] (1)

For proof see Paz (1971, p. 22).

Remark. It follows from the above theorem that for a given IS \( \mathcal{A} \) there are only finitely many other initial vectors \( \pi \) equivalent to \( \pi \). This follows from the fact that any such vector must have nonnegative integral entries with their sum equal to the sum of the entries of \( \pi \). On the other hand, all the vectors \( \pi \) equivalent to \( \pi \) can be found by using Eq. (1).

Let \( \mathcal{A} \) be an IS and let \( \xi^i(\sigma) \) be the \( i \)th row (assumed here to be a nonzero row) in a matrix \( A(\sigma) \) for some \( \sigma \). Let \( \xi' \) be an integral vector such that \( \xi^i(\sigma) H^\mathcal{A} = \xi' H^\mathcal{A} \) and let \( \mathcal{A}' \), be an IS derived from \( \mathcal{A} \) by replacing the row \( \xi^i(\sigma) \) of \( A(\sigma) \) with the row \( \xi' \) and replacing \( \pi \) with an equivalent initial integral vector \( \pi' \) (with respect to \( \mathcal{A} \)). We have the following.

THEOREM 3. The IS \( \mathcal{A} \) and \( \mathcal{A}' \), as above are equivalent.

For proof see Paz (1971, p. 23).
c. Reduction Theorems

**Definition.** A state \( i \) of an IS \( \mathcal{A}_n \) is accessible if there is a word \( x \in \Sigma^* \) such that \( \pi(x)_i > 0 \) (where \( \pi(x)_i \) denotes the \( i \)th entry of \( \pi(x) \)).

**Theorem 4.** If a state \( i \) of an \( n \)-state IS is accessible then it is accessible by a word of length \( \leq n \). The set of accessible states can effectively be found.

*Proof.* Trivial.

**Theorem 5.** If there is a state \( i \) of an \( n \)-state IS \( \mathcal{A}_n \) which is not accessible then the given IS can be reduced to an \( n - 1 \)-state equivalent IS.

*Proof.* Delete the \( i \)th entry in \( \pi \) (which must be zero); delete the \( i \)th columns and rows in the matrices \( A(\sigma) \) (if \( j \) is an accessible state then the \( j \)th row in \( A(\sigma) \) must have a zero entry in its \( i \)th column) and reduce \( \eta \) to an \((n - 1)\)-dimensional vector.

**Theorem 6.** Let \( \mathcal{A}_n \) be an \( n \)-state IS such that its \( H^\sigma \) matrix has two equal rows, then \( \mathcal{A}_n \) can be reduced to an \((n - 1)\)-state equivalent IS.


**Example.** Consider the following IS \( \mathcal{A}_n \):

\[
\pi = [1, 1, 1, 1] \quad \Sigma = \{\sigma_1, \sigma_2\}
\]

\[
A(\sigma_1) = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 2 \\ 1 & 0 & 0 & 1 \end{bmatrix} \quad A(\sigma_2) = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 \end{bmatrix} \quad \eta = \begin{bmatrix} 1 \\ 1 \end{bmatrix}
\]

Thus the first and fourth row of \( H \) are equal, and, therefore, one can find a 3-state equivalent IS \( \mathcal{A}_n' \):

\[
\pi' = (2 \ 1 \ 1) \quad \Sigma = \{\sigma_1, \sigma_2\}
\]

\[
A'(\sigma_1) = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 0 & 0 \\ 4 & 0 & 0 \end{bmatrix} \quad A'(\sigma_2) = \begin{bmatrix} 2 & 0 & 0 \\ 4 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \eta' = \begin{bmatrix} 1 \\ 1 \end{bmatrix}
\]
with
\[ H' = \begin{bmatrix} 1 & 2 \\ 1 & 0 \\ 1 & 4 \end{bmatrix}. \]

Although it is clear that no further reduction is possible using the previous Theorems 5 and 6 still the above IS is equivalent to the following \( \mathcal{A}'_{\pi} \)

\[ \pi'' = (0 \ 2 \ 2) \]
\[ A''(\sigma_1) = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 2 & 2 \end{bmatrix} \quad A''(\sigma_2) = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad \eta'' = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \]

that \( \mathcal{A}'_{\pi} \cong \mathcal{A}'_{\pi} \) follows from Theorem 3. The first state is not accessible in \( \mathcal{A}_{\pi} '' \), and, therefore, a 2-state equivalent IS \( \mathcal{A}_{\pi} \) can be found

\[ \tilde{\pi} = [2, 2] \quad \bar{A}(\sigma_1) = \begin{bmatrix} 0 & 0 \\ 2 & 2 \end{bmatrix} \quad \bar{A}(\sigma_2) = \begin{bmatrix} 2 & 2 \\ 0 & 0 \end{bmatrix} \quad \bar{\eta} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}. \]

Notice that the derivation of \( \mathcal{A}_{\pi} '' \) from \( \mathcal{A}_{\pi} \) was made possible by the fact that in \( H' \) the first row was a convex combination of the other two rows. Such a condition is, however, not sufficient, and there are other conditions deriving from the requirement that the resulting matrices and initial vector have only integral values which must be considered. These considerations lead to the following problem: Given an \( n \)-state IS, give an algorithm which will decide whether there exists an equivalent IS with less than \( n \) states and, if the answer is positive, will provide a procedure by which such an equivalent IS could be constructed.

d. General Reduction Problem

Let \( \mathcal{A}_{\pi} \) be a given IS and consider the infinite ordered set of vectors \( K^{\mathcal{A}} \) and the matrix \( L^{\mathcal{A}_{\pi}} \) as defined in the previous section. Denote by \( [K^{\mathcal{A}}] \), the infinite matrix whose \( i \)th column is the \( i \)th vector in \( K^{\mathcal{A}} \). Define the infinite matrix \( [K^{(\mathcal{A},\pi)}] \) as

\[ [K^{(\mathcal{A},\pi)}] = L^{\mathcal{A}_{\pi}} [K^{\mathcal{A}}]. \]

**Theorem 7.** Let \( \mathcal{A}_{\pi} \) and \( \mathcal{A}_{\pi}' \) be two IS over the same \( \Sigma \). \( \mathcal{A}_{\pi} \) is equivalent to \( \mathcal{A}_{\pi}' \) if and only if there exists an integral nonnegative matrix (i.e., with all its entries nonnegative integers) \( B^* \) such that its first row is \( \pi^* \) and

\[ B^*[K^{\mathcal{A}_{\pi}'}] = L^{\mathcal{A}_{\pi}} [K^{\mathcal{A}}]. \]
Assume first that \( \mathcal{A}_n \cong \mathcal{A}_n^* \). Then for any \( x_1, x_2 \in \Sigma^* \), \( \pi^*(x_1) \eta^*(x_2) = \pi(x_1) \eta(x_2) \). If the \( i \)th row in \( L_{\mathcal{A}_n} \) is \( \pi(x_i) \) choose the \( i \)th row in \( B^* \) to be \( \pi^*(x_i) \). (The first row in \( L_{\mathcal{A}_n} \) is \( \pi \) which will correspond to \( \pi^* \) in \( B^* \)). This implies (2) for \( \eta^*(x_2) \) and \( \eta(x_2) \) are corresponding columns in \( K^{\mathcal{A}_n} \) and \( K^* \), respectively.

Assume now that (2) holds for some matrix \( B^* \) with first row equal to \( \pi^* \), then the first row of the equation (2) has the form

\[
\pi^*[K]\pi^* = \pi[K],
\]

which implies that \( \pi^*[\eta](x) = \pi[\eta](x) \) for all \( x \in \Sigma^* \) so that the two IS are equivalent.

Let \( \mathcal{A}_n \) be a given IS and consider the matrix \( [K^{(\mathcal{A}_n, \pi)}] = L_{\mathcal{A}_n}[K] \) as above. It can be shown (see Paz (1971, p. 51)) that it is possible to effectively construct a matrix, to be denoted \( H^{(\mathcal{A}_n, \pi)} \), such that

1. The columns of \( H^{(\mathcal{A}_n, \pi)} \) are linearly independent vectors from the set \( K^{(\mathcal{A}_n, \pi)} \) and all other vectors in the set are a linear combination of them.
2. The columns in \( H^{(\mathcal{A}_n, \pi)} \) are the first columns in \( [K^{(\mathcal{A}_n, \pi)}] \) satisfying the condition (1) above.

In fact, one can show that the columns of \( H^{(\mathcal{A}_n, \pi)} \) can be chosen to be vectors of the form \( L_{\mathcal{A}_n}[\eta](x) \) with \( |x| \leq n - 1 \) (given that \( \mathcal{A}_n \) is an \( n \)-state IS).

**(Theorem 8.)** Let \( \mathcal{A}_n \) be an IS over an alphabet \( \Sigma \), and let \( m \) be the number of columns in its \( H^{(\mathcal{A}_n, \pi)} \) matrix. No \( m^* \)-state IS \( \mathcal{A}_n^* \) over the same alphabet \( \Sigma \) with \( m^* < m \) can be equivalent to \( \mathcal{A}_n \).

**Proof.** If \( \mathcal{A}_n \cong \mathcal{A}_n^* \), then by (2) there exists a matrix \( B^* \) such that \( B^*[K] = L_{\mathcal{A}_n}[K] = [K^{(\mathcal{A}_n, \pi)}] \). But \( H^{(\mathcal{A}_n, \pi)} \) is a submatrix of \( [K^{(\mathcal{A}_n, \pi)}] \) with \( m \) independent columns. Therefore, \( [K^{(\mathcal{A}_n, \pi)}] \) must have \( m \) independent rows which implies by (2) that \( [K] \) has at least that many rows.

Let \( \mathcal{A}_n \) be a given IS and consider the following equations with \( \Delta(\sigma) \) an unknown (not necessarily integral) matrix, for all \( \sigma \in \Sigma \),

\[
L_{\mathcal{A}_n} A(\sigma) = \Delta(\sigma) L_{\mathcal{A}_n},
\]

(This equation has exactly one solution which can be found effectively. This follows from the fact that the rows of \( L_{\mathcal{A}_n} \) are linearly independent while the rows of \( L_{\mathcal{A}_n} A(\sigma) \) being in the set \( G^{(\mathcal{A}_n)} \) depend on the rows of \( L_{\mathcal{A}_n} \).)
It follows that the equation

\[ L^\sigma \pi A(\sigma) H^\sigma = \Delta(\sigma) L^\sigma \pi H^\sigma \]  

has at least one solution.

Let \( \Delta(\sigma) \) be any solution of Eq. (4). Then \( \Delta(\sigma) \) satisfies also the following equation for all \( x \in \Sigma^* \)

\[ L^\sigma \pi \eta(\sigma x) = L^\sigma \pi A(\sigma) \eta(x) = \Delta(\sigma) L^\sigma \pi \eta(x). \]  

This follows from the fact that the vectors \( \eta(x) \) are linear combinations of the columns of \( H^\sigma \).

Let \( \eta(\pi, x) \) denote the vector \( L^\sigma \pi \eta(x) \). Thus, \( \eta(\pi, x) \) is the vector corresponding to the word \( x \) in the matrix \([K(\sigma, \pi)] = L^\sigma \pi [K^\sigma]\). We have the following.

**Theorem 9.** Given an IS \( \mathcal{A}_\pi \), there are matrices \( \Delta(\sigma) \), for each \( \sigma \in \Sigma \), (not necessarily integral) such that

\[ \Delta(\sigma)[K(\sigma, \pi)] = [\eta(\pi, \sigma), \eta(\pi, \sigma x_1), \eta(\pi, \sigma x_2), \ldots], \]  

where \( \lambda, x_1, x_2, \ldots \) is the fixed enumeration of the words in \( \Sigma^* \).

Notice that while \( \Delta(\sigma) \) may have nonintegral entries both \([K(\sigma, \pi)]\) and the infinite matrix on the right side of Eq. (6) have only integral nonnegative values.

We are now able to prove the following.

**Theorem 10.** Let \( \mathcal{A}_\pi^* \) and \( \mathcal{A}_\sigma^* \) be two IS over the same alphabet \( \Sigma \). \( \mathcal{A}_\pi^* \) is equivalent to \( \mathcal{A}_\sigma^* \) if and only if there exists an integral nonnegative matrix \( B^* \) with first row equal to \( \pi^* \) such that the following equations hold true:

\[ B^* \eta^* = \eta(\pi, \lambda), \quad B^* A^*(\sigma) H^\sigma = \Delta(\sigma) B^* H^\sigma \quad \text{all } \sigma \in \Sigma, \]  

where \( \Delta(\sigma) \) are matrices as defined in (6).

**Proof.** We know already that the conditions of the Theorem are equivalent to the existence of a matrix \( B^* \) as required and satisfying Eq. (2); i.e.,

\[ B^*[K^\sigma^*] = [K(\sigma, \pi)]. \]
We will show that the condition (8) above is equivalent to the following:

\[ B^*\eta^* = \eta(\pi, \lambda) \quad \text{and} \quad B^*A^*(\sigma) \eta^*(x) = \Delta(\sigma) B^*\eta^*(x), \quad \text{all} \ x \in \Sigma^*, \quad (9) \]

with the same matrix \( B^* \).

Clearly, \( A^*(\sigma) \eta^*(x) = \eta^*(\sigma x) \) and by (8) \( B^*\eta^*(\sigma x) = \eta(\pi, \sigma x) \). On the other hand, by (8), we have that \( B^*\eta^*(x) = \eta(\pi, x) \) and it follows from (6) that \( \Delta(\sigma) \eta(\pi, x) = \eta(\pi, \sigma x) \). Thus, (8) \( \Rightarrow \) (9).

Assume now, that (9) holds true, then we prove by induction on the length of \( x \) that \( B^*\eta^*(x) = \eta(\pi, x) \). For \( x = \lambda \) (9) and (8) are identical.

For \( x = \sigma x' \) we have by (9) and by the induction hypothesis that \( B^*\eta^*(\sigma x') = B^*A^*(\sigma) \eta^*(x') = \Delta(\sigma) B^*\eta^*(x') = \Delta(\sigma) \eta(\pi, x') \). This implies by (6) that \( B^*\eta^*(\sigma x') = \eta(\pi, \sigma x') \) as required. Thus, (9) \( \Rightarrow \) (8).

We prove now that (9) is equivalent to (7) with same matrix \( B^* \). That (9) \( \Rightarrow \) (7) is trivial for the columns in \( H^{\omega \sigma}\) are of the form \( \eta^*(x) \) for some \( x \in \Sigma^* \). The converse is also easy, for any column of the form \( \eta^*(x) \) can be expressed as a linear combination of the columns of \( H^{\omega \sigma} \) by definition.

**Corollary 11.** Let \( \mathcal{A}_{m, n}^* \) and \( \mathcal{A}_n \) be two IS over the same alphabet \( \Sigma \). Let \( \mathcal{J}_{\omega \sigma}^* \) be a matrix whose columns are the columns in \( [K(\mathcal{A}_{m, n}^*)] \) corresponding to the same words in \( \Sigma^* \) as the columns in \( H^{\omega \sigma} \) and in the same order. \( \mathcal{A}_{m, n}^* \) is equivalent to \( \mathcal{A}_n \) if and only if there exists a nonnegative integral matrix \( B^* \) whose first row equals \( \pi^* \) and such that

\[ B^*\eta^* = \eta(\pi, \lambda) \quad \text{and} \quad B^*H^{\omega \sigma}(\sigma) = \mathcal{J}_{\omega \sigma}^*(\sigma), \quad \text{for all} \ \sigma \in \Sigma, \quad (10) \]

\[ H^{\omega \sigma}(\sigma) = A^*(\sigma) H^{\omega \sigma} \quad \text{and} \quad \mathcal{J}_{\omega \sigma}(\sigma) = \Delta(\sigma) \mathcal{J}_{\omega \sigma}. \]

Notice that the columns of \( H^{\omega \sigma}(\sigma) \) are columns in \( [K(\mathcal{A}_{m, n}^*)] \), and, therefore, all the entries in \( H^{\omega \sigma}(\sigma) \) are nonnegative and integral. Similarly, the columns in \( \mathcal{J}_{\omega \sigma}(\sigma) \) are the columns in \( [K(\mathcal{A}_n)] \), corresponding to the same words in \( \Sigma^* \) as the columns in \( H^{\omega \sigma}(\sigma) \), and, therefore, the entries in \( \mathcal{J}_{\omega \sigma}(\sigma) \) are also nonnegative and integral.

**Corollary 12.** Let \( \mathcal{A}_{m, n}^* \) be an \( n \)-state IS and let \( \mathcal{A}_n \) be another IS over the same alphabet \( \Sigma \). Then \( \mathcal{A}_{m, n}^* \) is equivalent to \( \mathcal{A}_n \) if and only if there exists a nonnegative integral matrix \( B^* \) whose first row equals \( \pi^* \) and such that

\[ B^*[K(\mathcal{A}_{m, n}^*)(n)] = [K(\mathcal{A}_n)(n)], \quad (11) \]

where \( [K(\mathcal{A}_{m, n}^*)(n)] \) is the matrix whose columns are the vectors \( \eta^*(x) \) with \( |x| \leq n \) and similarly for \( [K(\mathcal{A}_n)(n)] \).
Proof. \( K^{\omega, \tau}(n) \) includes the vector \( \eta \) and it follows from Theorem 1 that the columns of \( H^{\omega, \tau}(\sigma) \) are columns in \([K^{\omega, \tau}(n)]\) so that (11) implies (10). On the other hand, we have as an immediate consequence of Corollary 11, that (10) implies (11). 

Remark. One can use now the above Corollary 12 and prove the decidability of the equivalence problem of two IS. We shall postpone, however, this problem and discuss it in a later section of this paper where an easier algorithm will be suggested for it.

We are now able to settle the minimization problem for IS. 

Theorem 13. Given an \( n \)-state IS \( \mathcal{A}_n \), there exists an effective algorithm which will construct another equivalent \( m \)-state IS \( \mathcal{A}_m^{**} \) with \( m < n \), if such an \( \mathcal{A}_m^{**} \) exists, or will decide that no such \( \mathcal{A}_m^{**} \) exists. 

Proof. We shall exhibit an algorithm which will perform the required task. Each step of the algorithm will be followed by an explanation if necessary. We shall need the following notation.

Let \( \xi_1, \xi_2, \ldots, \xi_k \) be a set of \( n \)-dimensional vectors then \( \varphi(\xi_1, \xi_2, \ldots, \xi_k) \) denotes the minimal hypersphere in \( n \)-space with center at origin including the point vectors \( \xi_1, \ldots, \xi_k \) in its interior or on its boundary. If \( U \) is a matrix whose rows are \( \xi_1, \ldots, \xi_k \) then, by definition, \( \varphi(U) = \varphi(\xi_1, \ldots, \xi_k) \).

Algorithm for Theorem 13. Step 1. Given the IS \( \mathcal{A}_n \), let \( t \) be the number of columns in the matrix \( H^{(\omega, \tau)} \), and let \( n \) be the number of states of \( \mathcal{A}_n \). Set \( m = t \).

Step 2. If \( m = n \) stop. There is no \( \mathcal{A}_m^{**} \) with less than \( n \) states and equivalent to \( \mathcal{A}_n \) (for \( m = t \) this follows from Theorem 8). Otherwise, go to the next step.

Step 3. Construct the matrix \([K^{(\omega, \tau)}(m)]\) and let \( \rho \) be the number of its rows. (It is clear that \( t \leq \rho \leq n \).)

Step 4. Construct a matrix \( B^* \) with \( \rho \) rows and \( m \) columns such that:

(a) All its entries are nonnegative integers.

(b) The sum of its columns is equal to the column vector \( \eta(\tau, \lambda) \).

(c) The matrix \( B^* \) has not been used in a previous application of step 4 of the algorithm.

If no such \( B^* \) matrix can be found then set \( m = m + 1 \) and go to Step 2, else, go to the next step.
Explanation. The matrix $B^*$ as constructed in Step 4 is intended to be the matrix satisfying Eq. (10) in Corollary 11, which implies the conditions (a) and (b). The third condition (c) is inserted here for the case where the algorithm will come back to Step 4 after going through other steps. It is clear that there are only finitely many matrices $B^*$ satisfying the conditions (a) and (b) for fixed $m$. Therefore, because of condition (c), the algorithm will pass through Step 4 only finitely many times before changing the parameter $m$.

Step 5. If the chosen matrix $B^*$ has no column with all its entries zero entries, then go to the next step. Otherwise, go to Step 10.

Step 6. Construct a matrix $U$ with $m$ rows and same number of columns as the matrix $[K^{(\mathcal{A}, \pi)}(m)]$, with all its entries nonnegative integers, with all its rows (when considered as point vectors) in the interior or on the boundary of $\varphi([K^{(\mathcal{A}, \pi)}(m)])$, with first column equal to $\eta$, such that $U$ has not been used in a previous application of Step 6, and such that $U$ satisfies the equation $B^*U = [K^{(\mathcal{A}, \pi)}(m)]$. If no such $U$ matrix can be found, then go to Step 4. Otherwise, go to the next step.

Explanation. The $U$ matrix is intended to be the $[K^{\mathcal{A}^*}(m)]$ matrix satisfying Eq. (11) in Corollary 12. According to that equation, the rows of $[K^{(\mathcal{A}, \pi)}(m)]$ must be integral nonnegative combinations of the rows of $U$ and every row of $U$ must be used in the construction of some row of $[K^{(\mathcal{A}, \pi)}(m)]$ (the matrix $B^*$ has no all-zero columns by Step 5). It is, thus, clear that no point-vector outside $\varphi([K^{(\mathcal{A}, \pi)}(m)])$ can participate in the formation of the rows of $[K^{(\mathcal{A}, \pi)}(m)]$. This implies that there are only finitely many matrices $U$ satisfying the conditions in Step 6 so that Step 6 will be used only finitely many times before changing the matrix $B^*$.

Step 7. Let the columns in $U$ corresponding to the columns $\eta(\pi, x)$ in $[K^{(\mathcal{A}, \pi)}(m)]$ be denoted by $\eta^*(x)$ where the same argument $x$ occurs in both vectors if they are in the same place. Choose a maximal set of linearly independent column vectors in $U$ such that the vectors chosen are the first vectors, according to their order in $U$, satisfying the required property (maximal linearly independent set). Denote the matrix whose columns are the above chosen columns ordered according to their original order in $U$ by $H^{\mathcal{A}^*}$ (this step relies on Theorem 1). Finally, construct the matrix $H^{\mathcal{A}^*}(\sigma)$ as follows: For every column $\eta^*(x)$ in $H^{\mathcal{A}^*}$, let the corresponding column in $H^{\mathcal{A}^*}(\sigma)$ be the column $\eta^*(\sigma x)$.

Step 8. Solve the equations $A^*(\sigma) = H^{\mathcal{A}^*}(\sigma)$ for every $\sigma \in \Sigma$, subject to the condition that all the entries in $A^*(\sigma)$ be integral and nonnegative. If for
some $\sigma$ no solution can be found then go to Step 6. Otherwise, go to the next step.

Explanation. The first column in $H^\sigma$ is a vector with all its entries equal to 1 while the entries in $A^\sigma(\sigma)$ must be integral and nonnegative. It follows that there may be only finitely many matrices $A^\sigma(\sigma)$ satisfying the equation in step 8 so that this step is decidable.

**Step 9.** Let $\pi^*$ be the first row of $B^*$ then $(\pi^*, \{A^\sigma(\sigma)\}, \eta^*)$ is an $m$-state IS equivalent to the given one. (The reader will prove this easily on the basis of the previous theorems and corollaries.) Stop.

**Step 10.** (This step is applicable only if the chosen $B^*$ matrix in Step 4 has one or more zero columns. For the sake of simplicity, we assume here that $B^*$ has only one all-zero column, the last one. The other cases are dealt with similarly. Of course, $B^*$ cannot be an all-zero matrix.) Let $B^{*'}$ be the matrix derived from $B^*$ by deleting its last (all-zero) column. Construct a matrix $U'$ with $m - 1$ rows satisfying the equation $B^{*'}U' = [K(\sigma, \sigma)(m)]$ with all its entries nonnegative integers, with all its rows (when considered as point vectors) in the interior or on the boundary of $\varphi([K(\sigma, \sigma)(m)])$ and with all entries in its first column equal to 1. Let the columns in $U'$ corresponding to the columns $\eta(\sigma, x)$ in $[K(\sigma, \sigma)(m)]$ be denoted $\eta^*(x)$ as in Step 7. Let $U'(m - 1)$ be the submatrix of $U'$ with columns corresponding to words $x$ with $|x| \leq m - 1$. Construct the matrix $U'(m - 1)(\sigma)$ as follows: for every column $\eta^*(x)$ in $U'(m - 1)$ let the corresponding column in $U'(m - 1)(\sigma)$ be the column $\eta^*(\sigma x)$. Finally, expand the matrix $U'$ to a matrix $U$ with $m$ rows as follows: The first $m - 1$ rows of $U$ are as in $U'$. The subvector of the last row of $U$ which belongs to columns corresponding to words $x$ with $|x| \leq m - 1$ (considered as a point vector) is in the interior or on the boundary of $\bigcup_{\sigma \in \Sigma} \varphi(U'(m - 1)(\sigma))$. The entries in the last row of $U$ which belong to columns corresponding to words $x$ with $|x| = m$ are left free at this stage of the algorithm and will be fixed, if possible, at a later stage. The matrix $U$ above should be chosen so that it differs in its fixed entries, from any other matrix $U$ chosen in a previous application of Step 10 of the algorithm. If no such matrix $U$ can be found then go to Step 4. Otherwise, go to the next step.

Explanation. Assume that the equations $A^\pi(\sigma)U'(m - 1) = U'(m - 1)(\sigma)$ have solutions $A^\pi(\sigma)$, such that $A^\pi(\sigma)$ is an $(m - 1) \times (m - 1)$ nonnegative integral matrix, for every $\sigma \in \Sigma$. Then an $(m - 1)$-state equivalent IS to the given IS $A_\pi$ can be constructed (by Corollary 12 and by the fact that $B^{*'}U'(m) = [K(\sigma, \sigma)(m)]$). This is impossible for in this case the algorithm would have stopped with a positive answer at an earlier stage. One may
assume, therefore, that there is a \( \sigma \in \Sigma \) such that no \( (m-1) \times (m-1) \)
nonnegative and integral matrix \( A^*(\sigma) \) exists which solves the equation
\( A^*(\sigma) U'(m-1) = U'(m-1)(\sigma) \). One must, therefore, expand this
equation to the equation \( A^*(\sigma) U(m-1) = U(m-1)(\sigma) \) with \( U(m-1) \)
having \( m \) rows, and try to solve this equation for an \( m \times m \) matrix \( A^*(\sigma) \)
having nonzero (and integral) entries, in that part of its last column corre-
spending to first \( m-1 \) rows. This implies that the last row of \( U(m-1) \)
must be in the interior or on the boundary of \( \bigcup_{\sigma \in \Sigma} \varphi(U'(m-1)(\sigma)) \) as
required. It is easily seen that the number of possible matrices \( U \) as constructed
in Step 10 for fixed \( m \) is finite.

**Step 11.** From the matrices \( U(m-1) \) and \( U(m-1)(\sigma) \) as constructed
in the previous step, construct the matrices \( H^\sigma \) and \( H^\sigma(\sigma) \), respectively,
as in Step 7. Some, but not all, entries in the last row of \( H^\sigma(\sigma) \) may not be
fixed yet. For example, the first column of \( H^\sigma(\sigma) \) (for any \( \sigma \)) has the form
\( \eta^\sigma(\sigma) \) which is a column in \( U(m-1) \) provided that \( m \geq 2 \), and this will
always be assumed (the other case is trivial). Thus, all the entries in the first
column of \( H^\sigma(\sigma) \) are fixed, for any \( \sigma \in \Sigma \).

**Step 12.** Solve the equations
\[
A^*(\sigma) H^\sigma = H^\sigma(\sigma)
\]
for \( m \times m \) matrices \( A^*(\sigma) \) with nonnegative and integral entries. If no such
solution exists then go to Step 10. Otherwise, go to step 9.

**Explanation.** The matrix \( A^*(\sigma) \) must have nonnegative and integral
entries and the sum of its columns must equal the first column of \( H^\sigma(\sigma) \)
(which is fixed). This implies that there are only finitely many possible
solutions to the equations in Step 12 which can be enumerated and checked
one after another. This observation (which is true also for the step 8 in the
algorithm) leads to the following.

**Corollary 14.** Given an \( n \)-state IS \( \mathcal{A}_n \), there are finitely many equivalent
\( m \)-state IS \( \mathcal{A}_{m^*} \) to it for any fixed \( m \) (including the case where there is no \( m \)-state
equivalent IS for the given one).

**Remark.** If a solution to the equations in Step 12 can be found such that
it fits the fixed entries in \( H^\sigma(\sigma) \), then the free entries in those matrices (and
also in \( U(m) \)) will be fixed by that solution.

**Remark.** The above algorithm is not optimal and many improvements
are possible.
e. Representability of Integral Word Functions

The following problem will be considered in this section. Given an integral word function \( f \) over an alphabet \( \Sigma \), \( f : \Sigma^* \rightarrow \mathbb{N} \) (where by "given" we understand that there is an effective procedure by which the values \( f(x) \) can be computed in finitely many steps for each \( x \in \Sigma^* \)). Is the given function representable in the form \( f = f^\mathcal{A}_n \) where \( \mathcal{A}_n \) is an IS?

**Definition 6.** Let \( f: \Sigma^* \rightarrow \mathbb{N} \) be an integral word function. Let \( \lambda, x^1, x^2, \ldots \) be a length preserving enumeration of the words in \( \Sigma^* \) (i.e., if \( |x^i| < |x^j| \) then \( i < j \)) and let \( \mathcal{H}(f) \) be the infinite matrix whose \( i - j \) entry is \( f(x^ix^j) \). The rank of \( f \cap (r(f)) \) is defined as the rank of the matrix \( \mathcal{H}(f) \); i.e., the maximal number of linearly independent rows (or columns) in it. Notice that \( r(f) = \text{rank } \mathcal{H}(f) \) may assume an infinite value.

The following theorems can now be proved (the reader is referred to Paz (1971, p. 134) for proofs of similar theorems).

**Theorem 15.** If \( f = f^\mathcal{A}_n \) where \( \mathcal{A}_n \) is an \( n \)-state IS then \( f \) has finite rank and \( r(f) \leq n \).

**Theorem 16.** If \( f \) is a given integral word function such that \( r(f) \leq n \), then a "pseudo integral" sequential system \( \mathcal{A}_n \) (meaning that the matrices \( A(\sigma) \) and vectors \( \pi \) and \( \gamma \) are not necessarily nonnegative and integral, but the function \( f^\mathcal{A}_n \) has only nonnegative and integral values) with number of states \( \leq n \) and such that \( f = f^\mathcal{A}_n \), can be found.

In addition to the above two theorems, one can also prove the following additional theorem which is peculiar to the integral nonnegative case (and is not true, in general).

**Theorem 17.** Let \( f \) be a given integral word function such that \( r(f) \leq n \). If \( f = f^\mathcal{A}^*_n \) with \( \mathcal{A}^*_n \), a true IS then the true IS \( \mathcal{A}^*_n \), (or an equivalent true IS) can be found.

**Proof.** Let \( \mathcal{A}_n \) be the pseudo IS satisfying \( f = f^\mathcal{A}_n \) as constructed in Theorem 16. For any \( m \), the matrix \( [K(\mathcal{A}_n, \pi)(m)] \) has only integral nonnegative values (the entries in that matrix are of the form \( \pi(x) \eta(y) = f^\mathcal{A}_n(xy) = f(xy) \) with \( x, y \in \Sigma^* \) although the matrices \( A(\sigma) \) and the vectors \( \pi \) and \( \gamma \) may have negative and nonintegral values.

Delete Step 2 from the algorithm proving Theorem 13 and apply the modified algorithm to the pseudo-IS \( \mathcal{A}_n \) above. It is easily seen that the
modified algorithm will search for a true \( m \)-state equivalent IS \( A^*_m \), with \( m \) growing larger and larger until such an equivalent IS is found and then stop. The algorithm will not stop and will run forever only if there is no true IS equivalent to the given pseudo-IS \( A_n \).

We conclude this section with a theorem giving a quite strong necessary condition for an integral word function \( f \) to be representable in the form \( f = f^{\sigma}_a \) with \( A_n \) an IS.

**THEOREM 18.** Let \( f \) be an integral word function. \( f \) is representable in the form \( f = f^{\sigma}_a \), with \( A_n \) an IS, only if for every \( x \in \Sigma^* \) there exists a set of numbers \( c_0, c_1, \ldots, c_{n-1} \) such that for every \( y, z \in \Sigma^* \) the following equality holds:

\[
f(yx^nz) = \sum_{i=1}^{n} c_{n-i} f(yx^{n-i}z).
\]

For proof see Paz (1971, p. 137).

\( f \). Closure Properties of ISF

**THEOREM 19.** Any word function \( f \) over an alphabet \( \Sigma \) of the form \( f(x) = c \) for all \( x \in \Sigma^* \) with \( c \) a nonnegative integer is an ISF.

**Proof.** Let \( \pi = [c] \), \( A(\sigma) = [1] \) for all \( \sigma \in \Sigma \) and \( \eta = [1] \). Then \( \pi A(x) y = c \) for all \( x \in \Sigma^* \).

**THEOREM 20.** If \( f^{\sigma}\pi \) and \( f^{\sigma}\pi' \) are ISF over the same alphabet \( \Sigma \) then so is \( cf^{\sigma}\pi + c\eta f^{\sigma}\pi' \) where \( c_1 \) and \( c_2 \) are nonnegative integers, and \( (c_1 f^{\sigma}\pi + c_2 f^{\sigma}\pi')(x) = c_1 f^{\sigma}\pi(x) + c_2 f^{\sigma}\pi'(x) \).

**Proof.** Let \( A^{\pi*} \) be defined as follows:

\[
\pi^* = [c_1 c_2 \pi] \quad A^{\pi*}(\sigma) = \begin{bmatrix} A(\sigma) & 0 \\ 0 & A'(\sigma) \end{bmatrix} \quad \eta^* = [\eta].
\]

It is easy to see that \( f^{\sigma}\pi^* = c_1 f^{\sigma}\pi + c_2 f^{\sigma}\pi' \).

**Corollary 21.** The equivalence problem for two IS over the same alphabet is decidable (i.e., one can decide effectively whether two IS are equivalent).

**Proof.** Let \( f^{\sigma}\pi \) and \( f^{\sigma}\pi' \) be two ISF Construct the IS \( A^{\pi*} \) with \( A^{\pi*}(\sigma) \) and \( \eta^* \) as in Theorem 20. Let \( \pi' = (\pi 0 \cdots 0) \) and \( \pi = (0 \cdots 0 \pi') \) where the number of zero entries in \( \pi' \) and \( \pi \), respec-
It is clear that $f^{\sigma^*} = f^{\sigma^*'}$ if and only if $\pi^1$ and $\pi^2$ are equivalent vectors for $\mathcal{A}^*$. Now use Theorem 2.

**Theorem 2.** If $f^{\sigma^*}$ and $f^{\sigma^*'}$ are ISF over the same alphabet $\Sigma$ then so is $f^{\sigma^*} \cdot f^{\sigma^*'}$, where $(f^{\sigma^*} \cdot f^{\sigma^*'})(x) = f^{\sigma^*}(x) \cdot f^{\sigma^*'}(x)$.

**Proof.** Define the IS $\mathcal{A}^*_{\sigma^*}$ with $\pi^* = \pi \otimes \pi'$, $A^*(\sigma) = A(\sigma) \otimes A'(\sigma)$ and $\eta^* = \eta \otimes \eta'$ where the operation $\otimes$ stands for Kronecker product of matrices; i.e., if $A = [a_{ij}]$ and $B = [b_{kj}]$ are (not necessarily square) matrices of order $m \times n$ and $p \times q$, respectively, then $A \otimes B = C = [c_{ik}] = [a_{ij}b_{kj}]$ by definition and the double indices $ik, jl$ of the elements of $C$ are ordered lexicographically

\[
\begin{align*}
ik &= 11, 12, \ldots, 1p, \ldots, m1, \ldots, mp; \\
jl &= 11, 12, \ldots, 1q, \ldots, n1, \ldots, nq;
\end{align*}
\]

One proves easily that $f^{\sigma^*}_{\pi^*} = f^{\sigma^*} \cdot f^{\sigma^*'}$ (see Paz (1971, pp. 101, 147)).

**Theorem 23.** Let $f^{\sigma^*}$ be an ISF and define the word function $g_y$ over the same alphabet $\Sigma$ with $y \in \Sigma^*$ as $g_y(x) = f^{\sigma^*}(yx)$. Then $g_y(x)$ is an ISF. Define the word function $g$ over the same alphabet $\Sigma$ as $g(\lambda) = 1$ and $g(\sigma x) = f^{\sigma^*}(x)$ for all $\sigma \in \Sigma$ and all $x \in \Sigma^*$. Then $g$ is an ISF.

**Proof.** Define the IS $\mathcal{A}^*_{\pi^*}$ such that $\mathcal{A}^* = \mathcal{A}$ and $\pi^* = \pi A(y) = \pi(y)$. Then $\pi^* A^*(\sigma) \eta^* = \pi A(\sigma) A(x) \eta = f^{\sigma^*}(yx)$. Thus, $g_y = f^{\sigma^*}$ as required.

Next, define the IS $\mathcal{A}_{\pi^*}$ as follows:

\[
\pi^* = \begin{pmatrix} 1 & 0 & \cdots & 0 \end{pmatrix}, \quad A^* = \begin{pmatrix} A(\sigma) \\ \vdash \\ 0 \end{pmatrix}, \quad \eta^* = \begin{pmatrix} 1 \\ \vdash \\ 0 \end{pmatrix},
\]

where $\pi^*$, $A^* \sigma$ and $\eta^*$ are vectors and matrices of dimension $n + 1$ if the dimensions of the vectors and matrices of $A^*$ are $n$. Clearly, $g(x) = f^{\sigma^*}(x)$.

**g. Single Letter Case**

All the properties of IS and ISF proved so far, are true, of course, for the case where the alphabet $\Sigma$, over which the functions are considered, consists of a single letter. There are, however, some additional properties peculiar to the single letter case. These properties will be discussed in this section. Given a word function over a single letter alphabet $\Sigma = \{\sigma\}$, $f : \Sigma^* \to N$, we shall use the notation $f(n)$ for $f(\sigma^n)$ so that the function is considered as a function $f : N \to N$. 
THEOREM 24. For integral word-functions $f$ over a single letter alphabet the following 4 conditions are equivalent:

1. $f = f^{\pi}$ for some pseudo IS $\mathcal{A}_\pi$.
2. The infinite Hankel matrix of $f$, $H(f)$ such that its $ij$ entry equals $f(i + j)$ is of finite rank.
3. The generating function of the infinite series $\sum_{i=0}^{\infty} f(i) x^i$ is rational (i.e., there are two polynomials in $x$; $p(x)$ and $q(x)$, such that

$$p(x) = q(x) \sum_{i=0}^{\infty} f(i) x^i$$

where equality means that the coefficients of $x^i$ are the same in both sides of the equation).

4. There exists an integer $n$ $\leq$ the number of states of $\mathcal{A}_n$ and constants $c_0, c_1 \cdots c_{n-1}$, such that for every integer $m \geq 0$ the following difference equation holds true:

$$f(m + n) = c_{n-1} f(m + n - 1) + c_{n-2} f(m + n - 2) + \cdots + c_0 f(m). \quad (12)$$

For proof see Paz (1971, b).

Every one of the four aspects exhibited in the above theorem can be helpful in the study of ISF and the growth function represented by them. Thus, the generating function approach has been used extensively by Szilard (1972) while the first and second aspect are dealt with in this paper, in a more general context.

We would like to stress, here, also, the usefulness of the fourth condition which is exhibited in the following theorems.

THEOREM 25. The growth of an ISF over a single letter is either polynomial or exponential or a combination of polynomial and exponential growth.

Proof. The relation (12) considered as a difference equation, homogeneous with constant coefficients, has solutions of the types stated in the theorem only.

Remark. The general solution of the difference equation (12) depends on initial conditions and it may happen that the growth of a specific solution is polynomial for a particular set of initial conditions and the growth is exponential for another set of initial conditions. It can also happen in other cases that the growth is polynomial for any set of initial conditions. Those cases are worth mentioning when applications to biological growth are considered.
THEOREM 26. Let $f = f^\alpha$ be an ISF over a single letter such that for some integers $m$ and $n$, $f(m) = f(m + 1) = \cdots = f(m + n)$ but there is $i > n$ such that $f(m + i) \neq f(m)$ then $\mathcal{A}_\alpha$ has at least $n + 1$ states.

Proof. If $\mathcal{A}_\alpha$ has no more than $n$ states then (12) holds true with $n_1 \leq n$ constants. Insert the values $f(m + n_1) = f(m + n_1 - 1) = \cdots = f(m)$ into it for the given $m$ (and after cancelling the equal values) we get that
\[
\sum_{i=0}^{n_1-1} c_i = 1.
\]
Let $i$ be the first integer $i > n$ such that $f(m) \neq f(m + i)$ and insert now into (12) the values $f(m + i) \neq f(m + i - 1) = f(m + i - 2) = \cdots = f(m + i - n)$. We have
\[
f(m + i) = \sum_{j=1}^{n_1} c_{n_1-j} f(m + i - j) = f(m + i - 1) \sum_{j=1}^{n_1} c_{n_1-j} = f(m + i - 1)
\]
a contradiction.

COROLLARY 27. Let $f$ be a word function over a single letter alphabet such that for every integer $n$ there are integers $m$ and $i > n$ such that $f(m + i) \neq f(m + n) = f(m + n - 1) = \cdots = f(m)$ then $f$ is not an ISF.

Proof. By Theorem 26 any IS representing $f$ must have infinitely many states.

3. GROWTH FUNCTIONS OF LINDENMAYER SYSTEMS

a. Growth in DOL-Systems

We begin by defining the notions of a DOL-system and its growth function. A deterministic informationless Lindenmayer system or, shortly, a DOL-system is an ordered triple
\[
S = (\Sigma, v, \delta),
\]
where $\Sigma$ is a finite nonempty set (the alphabet), $v \in \Sigma^+$ (the axiom) and $\delta$ is a mapping of $\Sigma$ into $\Sigma^*$. ($\Sigma^*$ was defined before. $\Sigma^+$ is the set of all nonempty words over $\Sigma$.) By considering $\delta$ as a homomorphism, we define $\delta^0(w)$, for any $w \in \Sigma^*$. By definition, $\delta^0(w) = w$ and $\delta^i$ denotes the composition of $i$ copies of $\delta$, for $i \geq 1$. The language generated by the DOL-system $S$ is defined by
\[
L(S) = \{\delta^n(v) \mid n \geq 0\},
\]
and its *growth function* by

\[ f_\Sigma(n) = |\delta^n(\varepsilon)|, \quad n \geq 0, \]

where (as before) vertical bars denote the length of the word.

For \( \sigma \in \Sigma \), the pair \((\sigma, \delta(\sigma))\) is written \( \sigma \rightarrow \delta(\sigma) \) and called a *production*. Our system is *propagating* or, shortly, a PDOL-system if \( \delta \) is a mapping into \( \Sigma^+ \), i.e., \( \delta(\sigma) \neq \lambda \), for each \( \sigma \in \Sigma \). As usual, the system being an \( L \)-system means that rewriting happens in a parallel manner, i.e., each letter is rewritten at every step of a derivation. The system being an \( O \)-system means that rewriting is context-free, i.e., the individual letters (the "cells") do not communicate with each other. Finally, the system being deterministic means that, for each \( \sigma \in \Sigma \), there is exactly one production with \( \sigma \) on its left side.

The general theory of integral sequential word functions developed in Section 2 is directly applicable to the growth functions of DOL-systems. In fact, the latter correspond to the single letter case of word functions. The general case will be applied to DTOL-systems in Section 3c. The context-dependent DL-systems considered in Section 3b possess an entirely different theory of growth.

As an example, consider the PDOL-system

\[ S = (\{a, b\}, a, \{a \rightarrow b, b \rightarrow ab\}). \]

The consecutive values of \( f_S(n) \) in this case form the Fibonacci sequence 1, 1, 2, 3, 5, 8, 13, 21,...

The *growth equivalence problem* for the class of DOL-systems is the problem of deciding for any two DOL-systems whether or not their growth functions are the same. The growth equivalence problem for any class of deterministic \( L \)-systems is defined in the same way.

For DOL-systems, the problem of finding *growth equivalent axioms* is defined as follows. Given a DOL-system (13), to find all DOL-systems with the same growth function, \( \Sigma \) and \( \delta \) as (13). Clearly, the number of such systems is finite since the new axiom has to be of the same length as \( \varepsilon \). The cell minimization problem consists of finding, for any given DOL-system, a growth equivalent (i.e., having the same growth function) DOL-system with minimal cardinality of the alphabet. The following problem of realizing a given growth with a given number of cells is more general: Given any DOL-system \( S \) and an integer \( k \geq 1 \), to find all DOL-systems which are growth equivalent to \( S \) and whose alphabet consists of \( k \) letters. (Of course, there may be no such DOL-systems.) Finally, the problem of realizing a function \( g \), from nonnegative integers into nonnegative integers, as a growth function
consists of finding, for any such \( g \), a DOL-system \( S \) with \( g = f_S \), provided such a system \( S \) exists.

We will now study each of these problems, using the results established in Section 2.

For a DOL-system (13) with the alphabet \( \Sigma = \{a_1, \ldots, a_k\} \), we define the following matrices. The initial vector, \( \pi \), is the \( k \)-dimensional row vector such that its \( i \)th component equals the number of occurrences of the letter \( a_i \) in the axiom \( v \), for \( i = 1, \ldots, k \). The final vector, \( \eta \), is the \( k \)-dimensional column vector with all components equal to 1. The growth matrix, \( A \), is the \( k \)-dimensional square matrix whose \((i, j)\)th entry equals the number of occurrences of \( a_i \) in \( \delta(a_j) \), for \( i, j = 1, \ldots, k \). These matrices are introduced because from the point of view of growth the order of letters in \( v \) and in each \( \delta(a_i) \) is immaterial. The following theorem is a direct consequence of the definitions.

**Theorem 28.** For any DOL-system \( S \), its growth function can be expressed in the form

\[
    f_S(n) = \pi A^n \eta,
\]

where \( A^0 \) is the identity matrix \( I \). Furthermore, if \( m \) is the length of the longest right side of the productions then

\[
    f_S(n) \leq m^n |v|, \quad \text{for all } n \geq 0.
\]

The representation (14) reduces the theory of growth functions of DOL-systems to the theory of integral sequential word functions (single letter case). The inequality (15) can be replaced by the more detailed characterization in Theorem 25.

We now use Theorem 2 to solve the problem of finding growth equivalent axioms.

**Theorem 29.** An algorithm for finding all growth equivalent axioms consists in finding all solutions \( \pi^x \) to Eq. (1), where \( \pi^1 \) is the initial vector of the given DOL-system.

As an example, consider the PDOL-system with the axiom \( a^2b^3c^2 \) and productions \( a \rightarrow ab^2c^4, \ b \rightarrow a^2b^4c^8, \ c \rightarrow a^4b^8c^{16} \). Its representation in terms of \( \pi, A, \eta \) is

\[
    \pi = (2 \ 2 \ 3), \quad A = \begin{bmatrix} 1 & 2 & 4 \\ 2 & 4 & 8 \\ 4 & 8 & 16 \end{bmatrix}, \quad \eta = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.
\]
The $H$ matrix for it is

$$H = \begin{bmatrix} 1 & 7 \\ 1 & 14 \\ 1 & 28 \end{bmatrix}$$

The equation $\pi H = (xyz)H$ has only two solutions $(x, y, z)$:

$$(2, 2, 3) \quad \text{and} \quad (0, 5, 2).$$

The first corresponds to the original axiom. Hence, the only other growth equivalent axiom is $b^2c^3$.

The generating function of $f_S(n)$ is defined to be the formal sum

$$F_S(x) = \sum_{n=0}^{\infty} f_S(n) x^n.$$

**THEOREM 30.** For any DOL-system $S$, the generating function of its growth function equals $\pi(I - Ax)^{-1}\eta$. The growth equivalence problem for DOL-systems is solvable.

**Proof.** We note first that the matrix $I - Ax$ is nonsingular because the elements of its main diagonal are of the form $1 - ax$; whereas, the remaining elements are of the form $a'x$. The first sentence of the theorem now follows by the representation (14) and the matrix equation

$$(I - Ax)^{-1} = \sum_{n=0}^{\infty} A^n x^n.$$ 

The generating function thus obtained is of the form $p(x)/q(x)$, where $p$ and $q$ are polynomials with integer coefficients. For another DOL-system $S_1$ with the generating function $p_1(x)/q_1(x)$ for its growth function, $S$ and $S_1$ are growth equivalent if and only if $p(x) q_1(x) = q(x) p_1(x)$, where the equality sign denotes the identity of the polynomials. Hence, the second sentence of the theorem follows.

The same decision method for PDOL-systems has been given by Szilard (1972). Another method has been given by Doucet (1972). A further decision method results as a special case of Corollary 21. Note also that Theorem 30 gives a method of determining the growth function of any DOL-system.

By Theorem 13 and Corollary 14 (cf. also the proof of Theorem 17), we obtain the following results.
Theorem 31. The cell minimization problem for DOL-systems is solvable, and so is the problem of realizing a given growth with a given number of cells.

We mention another application of Corollary 14.

Theorem 32. For any DOL-system S and integer k, there is only a finite number of DOL-systems growth equivalent to S and having k letters in their alphabet.

The problem of realizing a function as a growth function has been studied extensively by Szilard (1972). His methods give the answer for the case of PDOL-realizations of polynomials. Theorem 17 gives the following general result.

Theorem 33. There is an effective procedure with the following properties. Given a function g (from nonnegative integers into nonnegative integers) and a finite upper bound n for the rank of g, the procedure will output a DOL-system whose growth function equals g, provided such a system exists. If there is no such system, the procedure will run forever.

The procedure of Theorem 33 does not work if no upper bound n is given. However, if g results from experiments with a finitary device, it is clear that such an upper bound exists.

In many cases the closure properties discussed in Section 2f will give more practical methods for realizing functions as growth functions. For instance, the growth function of the PDOL-system with the axiom a and productions $a \rightarrow ab$, $b \rightarrow b$ equals the function $n + 1$. If one wants to realize $(n + 1)^2$ as a growth function, then one simply takes the Kronecker products of the matrices of the given system, obtaining the PDOL-system with the axiom a and productions $a \rightarrow abcd$, $b \rightarrow bd$, $c \rightarrow cd$, $d \rightarrow d$. The new system realizes the growth $(n + 1)^2$ but it is not minimal in terms of the number of letters. (Kronecker products usually give systems with more cells than necessary.) However, one can always apply the cell minimization procedure.

Following Szilard (1972), we say that the growth in a DOL-system S is malignant if there is no polynomial $p(n)$ such that $f_S(n) \leq p(n)$, for all n. The following theorem is easily obtained from the results of Szilard (1972).

Theorem 34. There is an algorithm for deciding whether or not the growth in a DOL-system is malignant.

Whether or not the growth is malignant is determined by the difference equation (12) and its initial conditions. As we pointed out in Section 2g, it
may happen that the same productions give rise to both malignant and “normal” growth, for suitable choices of the axiom. Of course, it may also happen that the growth is malignant, no matter how we choose the axiom, and also that the growth is normal no matter how we choose the axiom.

b. Context-Dependent DL-Systems

We will now consider the case where the rewriting may depend on the context. The productions are now of the form

\[(b, a, c) \rightarrow w, \quad b, a, c \in \Sigma, \quad w \in \Sigma^*,\]

meaning that an occurrence of the letter \(a\) lying between \(b\) and \(c\) is rewritten as \(w\). If this occurrence of \(a\) is the first or last letter of the word under scan, the missing context is provided by a fixed letter \(g\), so-called input from the environment.

Formally, a deterministic context-dependent Lindenmayer system or, shortly, a D2L-system is an ordered quadruple \(S = (\Sigma, v, g, \delta)\), where \(\Sigma\) and \(v\) are as in the definition of a DOL-system, \(g \in \Sigma\) and \(\delta\) is a mapping of the Cartesian power \(\Sigma^3\) into \(\Sigma^*\). If \(\delta\) is a mapping into \(\Sigma^+\), the system is termed propagating or a PD2L-system.

We now define a mapping \(\delta'\) of \(\Sigma^*\) into \(\Sigma^*\). For \(w = a_1 \cdots a_n\), where \(n \geq 2\) and each \(a_i\) is a letter,

\[\delta'(w) = \delta(g, a_1, a_2) \delta(a_1, a_2, a_3) \cdots \delta(a_{n-2}, a_{n-1}, a_n) \delta(a_{n-1}, a_n, g).\]

(Juxtaposition on the right side denotes the catenation of words.) For \(w = a_1 \in \Sigma\), \(\delta'(w) = \delta(g, a_1, g)\). Finally, \(\delta'(\lambda) = \lambda\). The language generated by \(S\) is now defined by

\[L(S) = \{(\delta')^n(v) \mid n \geq 0\}\]

and its growth function by

\[f_S(n) = |(\delta')^n(v)|, \quad n \geq 0.\]

A D2L-system is a D1L-system if and only if one of the following conditions holds: (i) for all letters \(a, b, c, d\), \(\delta(a, b, c) = \delta(a, b, d)\), or (ii) for all letters \(a, b, c, d\), \(\delta(a, b, c) = \delta(d, b, c)\). Thus, the numbers 0, 1, 2 in the definition of L-systems mean, respectively, that rewriting happens in a context-free, one-sided context-sensitive or two-sided context-sensitive manner. (As regards cells in filamentous organisms, the three alternatives mean, respectively, that individual cells do not communicate, or a cell may communicate...
with its neighbor which either is always the one on the left or always the one on the right or, finally, that a cell may communicate with both of its neighbors.)

As an example, consider the PD2L-system

$$S = (\{a, b, c, g\}, ba, g, \delta),$$

where $\delta$ is defined by

$$\begin{align*}
\delta(b, a, x) &= b, & \text{for all } x \neq c, \\
\delta(x, a, c) &= c, & \text{for all } x \neq b, \\
\delta(x, b, g) &= ac, & \text{for all } x, \\
\delta(x, b, y) &= a, & \text{for all } x \text{ and } y \text{ such that } y \neq g, \\
\delta(a, c, x) &= a, & \text{for all } x, \\
\delta(g, c, x) &= b, & \text{for all } x, \\
\delta(x, y, z) &= y, & \text{otherwise.}
\end{align*}$$

The sequence of words $$(\delta')^n(ba)$$ is

$$ba, ab, aac, aca, caa, baa, aba, aab, aaac, aaca, acaa, caaa,$$
$$baaa, abaa, aaba, aaab, aaac, aaaca, ...$$

and the first values of the growth function

$$2, 2, 3, 3, 3, 3, 3, 3, 4, 4, 4, 4, 4, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5, 5, 6,...$$

This example can be given the following interpretation. A filamentous organism grows only at its tail. Whenever growth has taken place, a message goes to the head which, in turn, sends back an instruction for another piece of growth. The more the organism grows, the more time it takes for these messages to get through.

By definition, the family of growth functions of context-dependent DL-systems includes the family of growth functions of informationless DL-systems. By the previous example and Corollary 27, we obtain the following theorem.

**Theorem 35.** There is a deterministic context-dependent Lindenmayer system whose growth function is not realizable by any DOL-system.

Our example for Theorem 35 is a PD2L-system but it can easily be replaced by a PD1L-system. In fact, Gabor Herman (personal communi-
tion) has constructed the following very slowly growing PD1L-system, where the lengths of the sequences of equal values grow exponentially. The axiom is $ad$, the input from the environment is $g$ and the productions are

$$(g, a) \rightarrow c, (c, a) \rightarrow b, (c, b) \rightarrow c, (c, d) \rightarrow ad, (x, c) \rightarrow a,$$

for all $x$.

Rewriting depends always on the left neighbor only and, thus, the right neighbor is missing from the left sides of the productions. (For instance, the first production means that an initial occurrence of $a$ is rewritten as $c$.) For all combinations not listed above, rewriting preserves the original letter. The first words in the sequence are now

$$ad, cd, aad, cad, abd, cbd, acd, abad, cbad, acad, cabd, abbd, cbbd, acbd, cacd, abaad,...$$

Note that growth can take place only after the messenger $c$ has reached $d$. This, in turn, can happen for words $cb'd$ only. In the above sequence, the distance between two words of this form grows exponentially.

Thus, the class of growth functions of PD1L-systems (resp. D1L-systems) properly includes the class of growth functions of PDOL-systems (resp. DOL-systems). It is an open problem whether or not there exists a PD2L-system (resp. D2L-system) whose growth function cannot be realized by any PD1L-system (resp. D1L-system). This problem can be further extended to concern $D(m, n)L$-systems, i.e., systems where the rewriting of each letter depends on $m$ of its left neighbors and on $n$ of its right neighbors. It has been shown by Rozenberg (1973a) that the families of languages generated by such systems form an infinite hierarchy. This does not imply that the families of growth functions also form an infinite hierarchy. Another open problem is to give a decision method for the growth equivalence problem of deterministic context-dependent Lindenmayer systems, perhaps only for a subclass of them such as PD1L-systems. No algorithm is known for deciding whether or not the growth in a context-dependent Lindenmayer system is malignant.

Comparing finite probabilistic and deterministic automata, it is well known that the former save states, i.e., there is a probabilistic automaton with two states which, for any $k$, accepts a language not acceptable by any deterministic automaton with fewer than $k$ states but acceptable by a deterministic automaton with $k$ states. A similar phenomenon is observed when comparing the growth functions of context-dependent and informationless $L$-systems. In the statement of the following theorem, a semi-PD1L-system means a PD1L-system without the axiom.
Theorem 36. There is a semi-PDL-system $S$ with three letters (including the input from the environment) such that, for each $k \geq 2$, there is an axiom $\psi_k$ and a PDOL-system $T_k$ with $k$ letters which satisfy both of the following conditions: (i) The growth function of $T_k$ cannot be realized by any PDOL-system with fewer than $k$ letters, (ii) The growth function of $T_k$ equals the growth function of $S_k$, the PD1L-system obtained from $S$ by adding the axiom $\psi_k$.

Proof. Define $S = \{a, b, c\}, \delta, \psi$, where for all letters $x$,

- $\delta(b, a, x) = b$, $\delta(a, b, x) = a$, $\delta(b, c, x) = a a$
- and $\delta(x, y, z) = y$, otherwise. Furthermore, for each $k \geq 2$, define

$$\psi_k = ba^{k-2}c,$$

$$T_k = \{a_1, \ldots, a_k\}, a_{k-1}a_2, \delta_k,$$

$$\delta_k(a_1) = a_1, \delta_k(a_k) = a_1a_1, \delta_k(a_i) = a_{i-1}, \text{ for } 2 \leq i \leq k - 1.$$

Then the following function $f$ is the growth function of both $T_k$ and $S_k$:

$$f(n) = \begin{cases} k & \text{for } n \leq k - 2, \\ k + 1 & \text{for } n > k - 2. \end{cases}$$

Condition (i) is satisfied because in any PDOL-system realizing $f$ the axiom must contain at least two distinct letters and, for all $i \leq k - 2$, the $i$th word must contain a letter which is not present in the $j$th word, for any $j < i$.

C. Growth Relations of DTOL- and OL-Systems

In systems considered so far, there is a unique sequence of words beginning with the axiom. We now consider cases where this condition is not satisfied, and, thus, we obtain a growth relation rather than a growth function.

A deterministic informationless Lindenmayer system with tables or, shortly, a DTOL-system is an ordered triple $S = (\Sigma, \psi, T)$, where $\Sigma$ and $\psi$ are as in the definition of a DOL-system and $T$ is a finite nonempty collection of mappings $t$ such that $(\Sigma, \psi, t)$ is a DOL-system for every $t \in T$. For each DOL-system $(\Sigma, \psi, t)$ thus obtained, we define the matrices $\pi, A(t)$ and $\eta$ as in Section 3a. The growth relation $R_S$ of $S$ is the binary relation defined as follows. For any $m, n \geq 0$, $R_S(m, n)$ holds if and only if either $m = 0$ and $n = \pi \eta$, or else $m > 0$ and there are elements $t_1, \ldots, t_m$ of $T$ such that

$$\pi A(t_1) \cdots A(t_m) \eta = n.$$
Two DTOL-systems $S$ (with matrices $\pi, A(t), \eta$) and $S'$ (with matrices $\pi', A'(t), \eta'$) are strongly growth equivalent if there is a one-to-one correspondence between the set of matrices $A(t)$ and the set of matrices $A'(t)$ such that, for any $m \geq 0$ and $t_1, \ldots, t_m$,

$$\pi A(t_1) \cdots A(t_m) \eta = \pi' A'(t_1') \cdots A'(t_m') \eta',$$

where $A'(t_i')$ is the matrix corresponding to $A(t_i)$. They are weakly growth equivalent if $R_S = R_{S'}$.

Intuitively, in a DTOL-system any element of $T$ (so-called "tables") may be applied to the word under scan, but different tables may not be mixed. The language generated by the system consists of all words obtained from the axiom in this fashion. A DOL-system can be viewed as a special case of a DTOL-system with only one table. If there are more than one tables, many words may be derived from the axiom in $m$ steps and, consequently, we have a growth relation rather than a growth function. By definition, strong growth equivalence of two systems implies that the systems have the same number of tables, i.e., the same degree of synchronization in the terminology of Rozenberg (1973b). In weak growth equivalence, only the lengths of the words are taken into account, not the number of different ways in which words of a given length may be derived.

The theory of integral sequential word functions is directly applicable to strong growth equivalence but not to weak growth equivalence. The results are summarized in the following theorem. The notions in the statement of the theorem are defined exactly as for DOL-systems, with the convention that equivalence means always strong growth equivalence. The theorem is obtained from Theorems 2 and 13 and Corollaries 14 and 21 in the same way as Theorems 29–32. It is to be emphasized that because only strong growth equivalence is considered, in each of the results one considers a family of DTOL-systems with the same degree of synchronization.

**Theorem 37.** There is an algorithm for finding all growth equivalent axioms for any DTOL-system. The growth equivalence problem for DTOL-systems is solvable. The cell minimization problem for DTOL-systems is solvable, and so is the problem of realizing a given growth with a given number of cells. For any DTOL-system $S$ and integer $k$, there is only a finite number of DTOL-systems growth equivalent to $S$ and having $k$ letters in their alphabet.

Finally, we consider OL-systems. An OL-system is defined as a DOL-system except that now $\delta$ is a mapping into the set of all nonempty finite subsets of $\Sigma^*$. One step in the rewriting process consists in replacing each
letter $a$ by some word in $\delta(a)$. Different occurrences of the same letter may be replaced by different words in $\delta(a)$ and, therefore, matrix approach will not be directly applicable. The growth relation $R_S$ of a OL-system $S$ is defined as follows. For any $m, n \geq 0$, $R_S(m, n)$ holds if and only if either $m = 0$ and $n$ is the length of the axiom, or else $m > 0$ and a word of length $n$ can be obtained from the axiom as the result of $m$ steps of rewriting. Two OL-systems or a DTOL-system and a OL-system are weakly growth equivalent if they have the same growth relation.

There are OL-languages which are not DTOL-languages, e.g. the language $\{a^{2n} \mid n \geq 0\}$ is generated by the OL-system $S$ with the axiom $aa$ and productions $a \rightarrow aa$ and $a \rightarrow \lambda$ but is not generated by any DTOL-system. However, the DTOL-system $S_1$ with the axiom $a_1a_2$ and tables $t$ such that $t(a_i) = w_i$, $i = 1, 2$, and the words $w_i$, independently of $i$, assume the values $a^2_1$, $a^2_2$, $a_1a_2$ and $\lambda$ is weakly growth equivalent to $S$. The same holds true also in general.

The idea in the proof of the following theorem is the same as in the example: Introduce new letters in such a way that if two occurrences of the same letter $a$ are rewritten differently according to the OL-system then in the DTOL-system they are replaced by two different letters $a_1$ and $a_2$.

**Theorem 38.** For any OL-system, there is a weakly growth equivalent DTOL-system.

**Proof.** Let the given OL-system be $S = (\Sigma, v, \delta)$. Without loss of generality, we assume that the following condition is satisfied for each letter $a$ in $\Sigma$; all letters occurring in some of the words in $\delta(a)$ are distinct among themselves and also different from $a$. (For if this is not the case originally, then we replace each $a$ in $\Sigma$ with sufficiently many new letters $a_{1}, \ldots, a_{k}$, referred to as descendants of $a$. The new set of productions consists of all productions obtained in the following way. The left side is a descendant of some letter $a$. The right side is obtained from a word in $\delta(a)$ by replacing every letter with one of its descendants in such a way that the new system satisfies the required condition. Since from the point of view of growth the descendants do not change anything, the new system is weakly growth equivalent to the original one.)

Thus, we assume that $S$ satisfies the condition mentioned above. Let $m$ be the maximum of the two numbers: the length of $v$ and the cardinality of $\Sigma$. Consequently, there are at most $m$ words in $\delta(a)$, for any $a$ in $\Sigma$. (This holds true also if $\lambda$ is among these words.) For each $a$ in $\Sigma$, introduce $m^2$ new letters $a_1, \ldots, a_{m^2}$, referred to as descendants of $a$. Let $\Sigma_1$ be the alphabet of all the new letters thus obtained. For a word $w$ over $\Sigma$, denote by $U(w)$ the (finite)
set of words over $\Sigma_1$ which are obtained from $w$ by replacing every letter with one of its descendants. (Different occurrences of the same letter may be replaced with different descendants.) Let $v_1 \in U(v)$ be such that different occurrences of the same letter are, in fact, replaced with different descendants. (Such a $v_1$ exists by the choice of the number $m$.) Let, finally, $T$ be the collection of mappings $t$ of $\Sigma_1$ into $\Sigma_1^*$, consisting of all mappings obtained in the following way. For each $a$ in $\Sigma$, denote by $U_1(a)$ the union of all sets $U(w)$, where $w$ ranges over the elements in $\delta(a)$. For each descendant $a_i$ of $a$, $t$ maps $a_i$ into some element in $U_1(a)$. Consider the DTOL-system $S_1 = (\Sigma_1, v_1, T)$. We claim that $S$ and $S_1$ are weakly growth equivalent.

In fact, if for some $i$ and $j$ we have $R_S(i, j)$ then we also have $R_S(i, j)$ because we only have to erase the indices indicating descendants to get the same growth. The converse implication follows from the subsequent observations: (i) in $v_1$ all letters are distinct, (ii) a step $w_1 \Rightarrow w_2$ in a derivation according to $S$ can be simulated by a step $w_1' \Rightarrow w_2'$ in a derivation according to $S_1$ in such a way that in $w_2'$ the letters of $w_2$ are indexed to take care of the next step of the derivation. (More specifically, (ii) can be established by induction on the length of the derivations in the following way. We make an inductive hypothesis $IH(i)$: Assume that $w$ is derived according to $S$ in $i \geq 0$ steps. Consider a word $w'$ obtained from $w$ by indexing the letters of $w$ in such a way that at most $m$ indices are used for each letter. Then one can derive according to $S_1$ in $i$ steps a word $w''$ obtained from $w'$ by permuting the letters. Using the inductive hypothesis $IH(i)$, the definition of $T$, and the fact that there are $m^2$ descendants for each letter, one immediately obtains $IH(i + 1)$.) Hence, $R_S = R_{S_1}$ and Theorem 38 follows.

The language generated by the DTOL-system $S$ with the axiom $a$ and two tables $(a \to a^3)$ and $(a \to a^3)$ is not generated by any OL-system. In the following theorem we show that no equivalent OL-system can be obtained even if attention is restricted only to growth relations.

**Theorem 39.** There is no OL-system which is weakly growth equivalent to the DTOL-system $S$ defined above. Consequently, the family of growth relations of DTOL-systems properly includes the family of growth relations of OL-systems.

**Proof.** Clearly, $R_S(m, n)$ holds if and only if $n = 2^{m-i}3^i$, for some $i$ such that $0 \leq i \leq m$. Assume that there is a OL-system $S_1$ such that $R_S = R_{S_1}$. For each $m_1$, there is an $m > m_1$ such that at the $m$th step of the rewriting process according to $S_1$ it is possible to replace an occurrence of a letter $a$ in a word $w$ by two words $w_1$ and $w_2$ of different lengths. (Otherwise, the
cardinalities of the sets

\[ R_m = \{ n \mid R_S(m, n) \}, \quad m = 0, 1, 2, \ldots \]

would be bounded.) Let \( u \) be greater than the greatest among the differences \( |x_1| - |x_2| \), where \( x_1 \) and \( x_2 \) are the right sides of some productions of \( S \) whose left sides coincide. Choose \( m_1 \) to satisfy \( 2^{m_1 - 1} > u \). Then, whenever \( m > m_1 \) and \( n_1 \) and \( n_2 \), \( n_1 > n_2 \), are such that \( R_S(m, n_1) \) and \( R_S(m, n_2) \) hold, we have \( n_1 - n_2 > u \). A contradiction now arises because the absolute value of the difference \( |w_1| - |w_2| \) is less than \( u \). This proves our theorem.

Clearly, the growth function or growth relation of any Lindenmayer system is bounded by a function \( \varphi(n) = ab^n \), where \( a \) and \( b \) are constants. Problems concerning malignant growth for systems more general than DOL-systems are left open.

Acknowledgment

Most of the work in this paper was done during the Open House in Unusual Automata Theory at DAIMI of Aarhus University, Denmark, January 10–28, 1972. The authors express their gratitude to the participants and organizers of this meeting, especially to Dr. Brian Mayoh.

Received: April 7, 1972

References

SZILARD, A. L. (1972), Growth functions of Lindenmayer systems, to be published.