ANNALES ACADEMIAE SCIENTIARUM FENNICAE

Series A

I. MATHEMATICA

444

ON DEFINITE AUTOMATA AND RELATED SYSTEMS

BY

MAGNUS STEINBY

HELSINKI 1969
SUOMALAINEN TIEDEAKATEMIA

doi:10.5186/aasfm.1969.444

Communicated 14 February 1969 by P. J. Myrberg and K. Inkeri

KESKUSKIRJAPAINO
HELSINKI 1969

PREFACE

I wish to express a deep acknowledgement to my inspiring teacher, Professor A. SALOMAA for the unfailing support he has given me in my work.

This dissertation has been written at the Institute of Theoretical Physics and the Institute of Mathematics, University of Turku. I am greatly indebted to the principals of these Institutes, Professor K. MANSIKKA and Professor K. INKERI, as well as to my colleagues there, for the good working conditions.

Thanks are in order to Lector A. T. LANDON, M. A., who has revised the English manuscript.

Turku, February, 1969.

MAGNUS STEINBY

Contents

Page

Chapter I. Definite automata and transition systems

Chapter II. Noninitial automata

Chapter III. Multidefinite automata

INTRODUCTION

The definite automata constitute one of the earliest and best known classes of finite automata. From the historical point of view, the concept has its origin in the paper [11] by McCulloch and Pitts, published in 1943. In this paper the loop-free neural nets were characterized in terms of logical calculus. As is well known, an automaton is definite iff it is realizable by a loopfree sequential network. The notion of definite events (here languages) was actually introduced 1956 by Kleene [9] in his famous paper, where he also gave a new formulation of the McCulloch-Pitts theory. The definition of definite languages has then been formulated in two equivalent ways in modern terminology.

Brzozowski [1] calls a language definite iff it can be given by a regular expression $E + X^*F$, where E and F denote finite languages. The simplicity of these definite expressions allows him to define a canonical representation for definite languages. This result (with an extension by Paz and Peleg [14]) is still one of the few partial solutions of the general problem of canonical forms of regular languages.

The other definition was given by Perles, Rabin and Shamir [15], who also introduced the related concept of definite transition tables (here transition systems). We adopt their definitions, but derive the fundamental properties needed using a special regular expression, called the k-form, for the definite language. In particular, we obtain the results of [15] that a reduced k-definite automaton has a k-definite transition system and at least $k + 1$ states by considering the derivatives of the k-form (§ 1).

The S.P. partitions of definite transition systems are considered in § 2. Thereby a new formulation and some generalizations of the contraction theory by Perles *et al.* are presented. The partitions and the corresponding homomorphic images of the transition system are then used in the second testing and analysis method given in § 3.

In § 4 we consider automaton partitions on the semigroup of input words. Definite automaton partitions are defined, and it is shown that any automaton partition defined by an S.P. partition of a definite transition system is definite. The special form of definite expressions makes it possible to develop an effective synthesis algorithm for definite languages from a method presented by LETITSEVSKI [10]. The method is based on the automaton partition theory and gives the minimal solution for any collection of definite languages. A bound for the number of refinements needed is given.

The remaining part of the first chapter deals with definite sequential machines and compositions of definite transition systems.

The relevance of definiteness in the theory of noninitial automata was noted by STARKE [19]. We treat noninitial automata in Chapter II starting with a slightly different definition of representation. In § 7 necessary and sufficient conditions under which a noninitial automaton represents a language are given. In § 8 a synthesis method giving the minimal solution for any collection of representable languages is developed through a suitable counterpart to the automaton partition theory.

A great number of different varieties and generalizations of definiteness appear in automaton theory. Such as the reverse definiteness by BRZOZOWSKI [1], ultimate-definiteness by PAZ and PELEG [14], and generalized definiteness by GINZBURG [4]. In the theory of probalistic automata we note the definite and quasi-definite tables, and quasi-definite languages introduced by PAZ [13], [14]. BRZOZOWSKI and SINGH [3] have considered definite asynchronous sequential circuits. Recently, SALOMAA [17] has introduced the notion of time-variant definite languages.

We begin Chapter III with a treatment of reverse definite automata and transition systems. Thereby a criterion for reverse definiteness is given.

In § 10 we introduce the multidefinite languages. The mode of generalization in question can be explained as follows. A sufficiently long word belongs to a given definite language iff it has a suffix of a specified length belonging to the language. A sufficiently long word belongs to a given multidefinite language iff it has some number of subwords, including a prefix and a suffix, of some specified lengths such that the concatenation of these subwords belongs to the language. We accomplish this generalization starting with the definition of definite languages by PERLES *et al.* From Theorem 23 it follows that the same family of languages could be obtained by starting with BRZOZOWSKIS definition.

Besides the definite languages, the family of multidefinite languages includes the reverse definite and the generalized definite languages. Through a canonical expansion we define a hierarchy of complexity for the multidefinite languages. Hereby the empty language and the universal language get the rank 0, all other definite and reverse definite languages the rank 1, and so forth.

In § 11 the corresponding generalization of definite transition systems is introduced and studied. It turns out that the rank of a multidefinite transition system cannot exceed 2. In § 12 some properties of multidefinite automata are presented. From Theorem 27 it follows that a reduced automaton has a multidefinite transition system iff it is generalized definite.

PRELIMINARIES

An alphabet is a nonempty finite set. For the alphabet X we denote by X^* the set of all finite sequences (*words*) of letters from X, including the empty word λ. The *union* or *sum* of two subsets (*languages*) of X^* U and V is denoted by $U + V$, their *product* is defined by $UV = \{p \mid p = uv , u \in U , v \in V\}$. Using the denotations $U^0 = \{\lambda\}$, $U^{i+1} = U^iU$ $(i = 0 , 1 , \ldots)$, the *iteration* of U is defined by

$$U = \{\lambda\} + U + U^2 + \ldots = \bigcup_{i=0}^{\infty} U^i .$$

The length of a word p is denoted by $L(p)$ and the *length* of a language $U \neq \emptyset$ is defined by

$$L(U) = \max \{L(p) \mid p \in U \} .$$

For $U = \emptyset$, we put $L(\emptyset) = -1$.

The unit languages $\{p\}$, with p in X^*, are written without parentheses: $\{p\} = p$. We use the symbol X to denote the sum of the letters in X, too. Then X^k $(k = 0 , 1 , 2 , \ldots)$ denotes the set of words of length k.

Let p, p' and p'' be words such that $p = p'p''$ and $L(p'') = k$ $(k = 0 , 1 , \ldots)$. Then we call p'' the *k-suffix* of p and denote it by $p|k$.

We give the definition of definite languages as it was formulated by PERLES, RABIN and SHAMIR [15].

Definition 1. Let k be a non-negative integer. The language D is *weakly k-definite* iff, for any word p satisfying $L(p) \geq k$, $p \in D$ iff $p|k \in D$.

When $k \geq 1$, D is *k-definite,* iff it is weakly k-definite but not weakly $(k - 1)$-*definite.* A language is 0-definite iff it is weakly 0-definite. The *degree* of D (denoted by deg$D$) is k iff D is k-definite.
A language is *definite* iff it is k-definite for some k.

It is easily seen that if D is weakly k-definite, then it is weakly k'-definite for any $k' \geq k$, and k''-definite for some unique $k'' \leq k$.

The definite languages can be introduced equivalently by calling a language definite iff it has a representation

$$D = Q + X^* R, \tag{1}$$

where Q and R are finite languages. We call (1) a *definite expression* for D iff Q and R are regular expressions denoting finite languages, where only the sum operation applied on words is used. If D is weakly k-definite, then it has a definite expression (1), where $R \subseteq X^k$ and $L(Q) < \mathrm{k}$. Suppose further that the terms in Q and R are arranged in order of increasing length and words of equal length in some given alphabetical order. This unique definite expression we call the *k-form* of D.

Note. We shall not bring forth explicitly the distinction between a regular expression and the language denoted by it. Thence we use the symbol of a regular expression to denote also the corresponding language.

The *canonical regular expressions* introduced by Brzozowski [1] provide another useful — and in general more compact — way of describing definite languages.

We call two languages U and V *almost equal* iff

$$|U - V| + |V - U| < \infty,$$

where $|W|$ denotes the cardinality of the set W.

If $D_1 = Q_1 + X^*R_1$ and $D_2 = Q_2 + X^*R_2$ are two definite expressions with $R_1 = R_2$, then the languages D_1 and D_2 are almost equal. Moreover the following lemma is obvious.

Lemma 1. Two weakly k-definite languages $Q_1 + X^*R_1$ and $Q_2 + X^*R_2$, given in their k-forms, are almost equal if and only if $R_1 = R_2$.

Hence almost equality divides the class of weakly k-definite languages into equivalence classes, each class corresponding to a unique subset of X^k.

An *(initial) automaton* over the alphabet X is an ordered quadruple

$$\mathfrak{A} = (S, \delta, s_0, F),$$

where $S \neq \emptyset$ is the set of states, $\delta : S \times X \rightarrow S$ the next-state function,

$s_0 \in S$ the initial state and $F \subseteq S$ the set of final (or representing) states. The pair (S , δ) is referred to as a *transition system,* and (S , δ , s_0) as an *initial transition system.* The automaton or transition system is called *finite* iff S is a finite set.

Following the common practice we extend the domain of δ to $S \times X^*$. We denote $\delta(s , p)$ by sp for any state s and word p. For a set $S_1 \subseteq S$ and word p we denote

$$\delta(S_1 , p) = \{s \in S | \, s_1p = s \text{ for some } s_1 \in S_1\} .$$

The language *represented* by $\mathfrak{A}$ is defined as $T(\mathfrak{A}) = \{p \in X^* | s_0p \in F\}$. The finite automaton is called *definite* or (weakly) k-definite iff $T(\mathfrak{A})$ is definite or (weakly) k-definite, respectively.

A language U is said to be *representable* in the initial transition system $A = (S , \delta , s_0)$ iff there exists a set $F \subseteq S$ such that $U = T(\mathfrak{A})$ for $\mathfrak{A} = (S , \delta , s_0 , F)$.

We shall frequently deal with partititons on a state set or on X^*. Let π be a partition on a nonempty set S. Identifying π with the corresponding equivalence relation on S we denote, for any $s , s' \in S$, $s\pi s'$ iff s and s' belong to the same class in π.

Let π^1 and π^2 be two partitions on S. We denote $\pi^1 \leq \pi^2$ and call π^1 a *refinement* of π^2 iff for each class $\pi^1_\alpha \in \pi^1$ there exists a class $\pi^2_\beta \in \pi^2$ such that $\pi^1_\alpha \subseteq \pi^2_\beta$. We write $\pi^1 < \pi^2$ iff the refinement is proper. The relation $\leq$ defines a partial ordering in the set of all partitions on S. It is easy to see that the set of all partitions on S forms a lattice relative to $\leq$. Hereby, the infimum and supremum of the partitions π^1 and π^2 are given by $\pi^1 \cdot \pi^2$ and $\pi^1 + \pi^2$, respectively, where the meet operation $+$ and the join operation $+$ are defined as follows.

1. For any $s , s' \in S$, $s(\pi^1 \cdot \pi^2)s'$ iff $s\pi^1s'$ and $s\pi^2s'$.

2. For any $s , s' \in S$, $s(\pi^1 + \pi^2)s'$ iff there exists a sequence

$$s_0 = s , s_1 , s_2 , \ldots , s_n = s'$$

of elements in S such that, for any $i = 0 , 1 , \ldots , n - 1$, $s_i\pi^1s_{i+1}$ or $s_i\pi^2s_{i+1}$.

The greatest element of this lattice is the *identity partition* $I = \{S\}$, and the least element is the *null partition* $0 = \{ \{s\} \mid s \in S\}$.

A partition on a state set is called a *state partition* and a partition on X^* an *input partition.*

CHAPTER I

DEFINITE AUTOMATA AND TRANSITION SYSTEMS

§ 1. Derivatives of definite languages

1.1. The *(left) derivative* of a language U with respect to a given word p is defined as the language

$$\partial_p U = \{q \in X^* \mid pq \in U\} .$$

The set of all distinct derivatives of a regular language U, here denoted by ∂U, corresponds in a well-known way to the state set of the minimal finite automaton representing U (cf. Brzozowski [2]). The special applicability of the synthesis method provided by this fact to definite automata was demonstrated by Brzozowski [1]. We shall make use of the derivatives to find some fundamental properties of definite automata in a simple way.

1.2. Let $D = Q + X^* R$ be a definite expression. Then for any letter x in X

$$\partial_x D = \partial_x Q + \partial_x R + X^* R . \tag{1.1}$$

The derivative is definite and by repeated use of (1.1) and the general rule $\partial_{pq} U = \partial_q(\partial_p U)$, we get for an arbitrary nonempty word $p = x_1 x_2 \ldots x_n$:

$$\begin{aligned}
\partial_p D &= \partial_{p|n-1} (\partial_{x_1} D) \\
&= \partial_{p|n-2} (\partial_{x_2} (\partial_{x_1} Q + \partial_{x_1} R + X^* R)) \\
&= \partial_{p|n-3} (\partial_{x_3} (\partial_{x_1 x_2} Q + \partial_{x_1 x_2} R + \partial_{x_2} R + X^* R)) \\
&\;\;\cdot \\
&\;\;\cdot \\
&\;\;\cdot \\
&= \partial_p Q + \partial_p R + \partial_{p|n-1} R + \ldots + \partial_{x_n} R + X^* R .
\end{aligned}$$

If we denote for any word p of length n and language U

$$\partial^p U = \bigcup_{i=1}^{n} \partial_{p|i} U$$

(especially $\partial^\lambda U = \emptyset$) , we get

$$\partial_p D = \partial_p Q + \partial^p R + X^* R \,. \tag{1.2}$$

Clearly (1.2) holds also when $p = \lambda$.

Suppose that D is weakly k-definite and that $Q + X^*R$ is its k-form. Then

$$L(\partial_p Q + \partial^p R) < k \,.$$

Hence the derivative is weakly k-definite and (1.2) gives its k-form when the words in $\partial_p Q + \partial^p R$ are ordered properly. Using Lemma 1 we get

Lemma 2. The derivatives of a weakly k-definite language D are weakly k-definite and almost equal to D .

It is to be noted that $\deg D$ is in general not retained in derivation. Similarly we get from formula (1.2) and Lemma 2

Lemma 3. Let D_1 and D_2 be two almost equal weakly k-definite languages. Then $\partial_p D_1 = \partial_p D_2$, for any word p satisfying $L(p) \geq k$. If $k > 0$ and $L(p) = k - 1$, then $\partial_p D_1$ and $\partial_p D_2$ differ at most by the empty word λ .

Theorem 1. A language is weakly k-definite if and only if, for any fixed word q of length k , all derivatives $\partial_q(\partial_p D)$ are equal.

Proof. Suppose that $\partial_q(\partial_{p_1} D) = \partial_q(\partial_{p_2} D)$ for all words p_1 and p_2 whenever $L(q) = k$. Let $p = p'(p|k)$ be an arbitrary word satisfying $L(p) \geq k$. Then by our assumption

$$\partial_p D = \partial_{p|k}(\partial_{p'} D) = \partial_{p|k}(\partial_\lambda D) = \partial_{p|k} D \,.$$

But this implies that $p \in D$ if and only if $p|k \in D$, and hence D is weakly k-definite. The converse part follows from Lemmas 2 and 3.

From Theorem 1 and Lemma 3 we get the following

Corollary 1.1. If $k > 0$, then any k-definite language has two derivatives differing only by the empty word λ .

Theorem 2. A k-definite language has at least $k+1$ different derivatives.

Proof. The case $k = 0$ is obvious. We suppose that $k > 0$. Let D be a k-definite language. For every integer $i = 0, 1, 2, \ldots$ we define an equivalence relation Ω^i on ∂D by the condition $\partial_u D \Omega^i \partial_v D$ iff $\partial_p(\partial_u D) = \partial_p(\partial_v D)$ for all $p \in X^i$.

Clearly $\partial_u D \Omega^i \partial_v D$ implies $\partial_u D\, \Omega^{i+1} \partial_v D$. Thus we have a nondescending chain of partitions on ∂D

$$0 = \Omega^0 \leq \Omega^1 \leq \ldots \leq \Omega^{k-1} \leq \Omega^k = I\,. \tag{1.3}$$

The last equality follows from Theorem 1. Clearly

$$\partial_u D \Omega^{i+1} \partial_v D \text{ iff } \partial_{ux} D \Omega^i \partial_{vx} D \text{ for any } x \text{ in } X\,.$$

Suppose that for some $i = 0, 1, \ldots, k-1$, $\Omega^{i+1} = \Omega^i$. We make the induction hypothesis that $\Omega^{i+n} = \Omega^i$. Then the following chain of equivalences is valid:

$$\begin{aligned}\partial_u D \Omega^{i+n+1} \partial_v D \text{ iff } & \partial_{ux} D \Omega^{i+n} \partial_{vx} D \text{ for any } x \text{ in } X\\ \text{iff } & \partial_{ux} D \Omega^i \partial_{vx} D \text{ for any } x \text{ in } X\\ \text{iff } & \partial_u D\, \Omega^{i+1} \partial_v D\\ \text{iff } & \partial_u D \Omega^i \partial_v D\,.\end{aligned}$$

Hence $\Omega^{i+1} = \Omega^i$ implies $\Omega^i = \Omega^{i+1} = \ldots = \Omega^{k-1} = \Omega^k = I$. But this contradicts Theorem 1. Hence all inclusions in (1.3) must be proper, and thus the number of derivatives must exceed k.

From this we obtain immediately the following result proved in another way by Perles *et al.* [15].

Corollary 2.1. A k-definite finite automaton has at least $k+1$ states.

§ 2. Definite transition systems

2.1. The definiteness of a transition system was defined in [15] as follows.

The transition system (S, δ) is *weakly k-definite* $(k \geq 0)$ iff for every word p, $L(p) = k$ implies $\delta(s, p) = \delta(s', p)$ for any pair of states s, s' in S. The notions of *k-definite* and *definite* transition system as well as the *degree* $\deg \delta$ of a definite transition system are derived from this in the way of Definition 1.

As a direct consequence of Theorem 1 we can state the following result due to PERLES, RABIN and SHAMIR [15].

Theorem 3. A reduced finite automaton is k-definite if and only if its transition system is k-definite.

2.2 Let π be a state partition of a transition system (S, δ). Following the terminology of [7] we call π a *partition with substitution property* or shortly an *S.P. partition* iff, for any states s and s' in S and any letter x,

$$s \, \pi \, s' \quad \text{implies} \quad \delta(s, x) \, \pi \, \delta(s', x) .$$

Hence a state partition π on S is an S.P. partition iff for each class $\pi_\alpha \in \pi$ and each letter x in X there exists a class $\pi_\beta \in \pi$ such that $\delta(\pi_\alpha, x) \subseteq \pi_\beta$. Clearly π_β is unique and thus a transition system

$$(S, \delta)_\pi = (\pi, \delta_\pi) ,$$

the *π-image* of (S, δ) can be introduced by defining

$$\delta_\pi(\pi_\alpha, x) = \pi_\beta \text{ iff } \delta(\pi_\alpha, x) \subseteq \pi_\beta ,$$

for all $\pi_\alpha, \pi_\beta \in \pi$ and $x \in X$.

Theorem 4. All π-images of a k-definite transition system are weakly k-definite. Hence all homomorphic images of a k-definite transition system are weakly k-definite.

The proof of this theorem is straightforward. The second proposition follows from the first because each homomorphic image of a transition system is isomorphic to the π-image, where π is the S.P. partition induced by the homomorphism.

2.3. We now introduce an important class of S.P. partitions of a transition system. The concepts defined in the following two definitions are generalizations of those (for $n = 1$) used in [15].

Definition 2. Two states s and s' of a transition system (S, δ) are *n-equivalent* $(n = 0, 1, \ldots)$ iff $\delta(s, p) = \delta(s', p)$ for any word p of length n.

Clearly this is for every n an equivalence relation on S corresponding to that introduced for derivatives of a given language in the proof of Theorem 2. We denote the relation and the corresponding partition on S by Ω^n.

Lemma 4. For every $n = 0, 1, \ldots$

(a) Ω^n is an S.P. partition,

(b) $\Omega^n \leq \Omega^{n+1}$, and

(c) if $\Omega^{n+1} = \Omega^n$, then $\Omega^{n+i} = \Omega^n$ for every $i = 0, 1, \ldots$.

Proof. The properties (a) and (b) follow easily from definitions, and (c) can be proven by a technique similar to that used in the proof of Theorem 2.

From Lemma 4 we get the following theorem, which includes the existence of 1-equivalent states proved by SIMON [18].

Theorem 5. The transition system (S, δ) is weakly k-definite if and only if $\Omega^k = I$. If the transition system is k-definite with $k > 0$, then

$$\Omega^0 < \Omega^1 < \ldots < \Omega^{k-1} < \Omega^k = I .$$

Hence there exists for every $i = 1, 2 \ldots, k$ a pair of i-equivalent states which are not $(i - 1)$-equivalent.

As noted in Lemma 4 the partitions Ω^n are S.P. partitions. We shall now show that the Ω^n-images can be constructed practically by means of the strong contractions used in [15].

Definition 3. The *nth* (*strong*) *contraction* (S_n, δ_n), $n \geq 0$, of a transition system (S, δ) is defined recursively as follows.

1. $(S_0, \delta_0) = (S, \delta)$.

2. Given (S_i, δ_i), $i = 0, 1, \ldots, (S_{i+1}, \delta_{i+1})$ is obtained from (S_i, δ_i) by identifying the states in each class of 1-equivalent states as one state.

The identifications in 2 can be performed by choosing from each set of 1-equivalent states a representative, removing the other states from the state set and replacing them in the remaining transition table by their representatives. Hence we may suppose that $S_{n+1} \subseteq S_n \subseteq S$ for every n. We suppose that the representatives are always chosen according to some prescribed rule. Then a *contraction operator* C performing the contractions can be introduced: $C(S, \delta) = (S_1, \delta_1)$ and Cs is the representative of the class of 1-equivalent states in (S, δ) to which s belongs. In general, we can write $C^n(S, \delta)$ for (S_n, δ_n), and $C^n s$ for the representative of s in S_n. Hereby C^0 is the identity operator. If $C^n s = C^n s'$, then we say that s and s' are *identified* in S_n.

Lemma 5. If $\delta(s, x) = s'$, then $\delta_n(C^n s, x) = C^n s'$ $(n = 0, 1, \ldots)$.

Proof. The proposition holds in the cases $n = 0$ and $n = 1$ per definitionem. Suppose that it holds for $n = i$, and that $\delta(s, x) = s'$. Clearly, $C^i s$ and $C^i s'$ are 1-equivalent to $C^{i+1}s$ and $C^{i+1}s'$, respectively, in (S_i, δ_i). Hence

$$\delta_i(C^{i+1}s, x) = \delta_i(C^i s, x) = C^i s'$$

by the induction hypothesis. The definition of (S_{+1}, δ_{i+1}) implies now that

$$\delta_{i+1}(C^{i+1}s, x) = C^{i+1}s' .$$

Theorem 6. Two states of the transition system (S, δ) are identified in the *nth* contraction (S_n, δ_n), $n = 0, 1, \ldots$, if and only if they are n-equivalent. Hence (S_n, δ_n) is the Ω^n-image of (S, δ).

Proof. The validity of the theorem for $n = 0$ is obvious. Suppose that it holds for $n = i$, $i = 0, 1, \ldots$, and consider the case $n = i + 1$.

First let s and s' be $(i + 1)$-equivalent states. Then $(sx)q = (s'x)q$, for any word xq in X^{i+1}. Hence sx and $s'x$ are identified in S_i, for any letter x in X. But this implies by Lemma 5 that the representatives of s and s' are 1-equivalent in (S_i, δ_i) and hence identified in (S_{i+1}, δ_{i+1}).

Suppose conversely that the states s and s' are identified in S_{i+1}. Then their representatives $\bar{s}$ and $\bar{s}'$ in S_i are 1-equivalent (possibly equal) in (S_i, δ_i), i.e. $\delta_i(\bar{s}, x) = \delta_i(\bar{s}', x)$ for any x. But on the other hand, $\delta_i(\bar{s}, x)$ and $\delta_i(\bar{s}', x)$ are the representatives of sx and $s'x$ in S_i. Hence by the induction hypothesis $sx\Omega^i s'x$ for every letter x. This implies $s\Omega^{i+1}s'$.

Hence each class of Ω^n has a unique representative in S_n. If s and s' are the representatives of the classes Ω_1^n and Ω_2^n, then $\delta(\Omega_1^n, x)$

$\subseteq \Omega_2^n$ implies $\delta_n(s , x) = s'$, and conversely. Thus (S_n , δ_n) is isomorphic to $(S , \delta)_{\Omega^n}$.

Combining Theorem 6 with Theorem 5 we get, successively, the following results presented or implicit in [15].

Corollary 6.1. The transition system (S , δ) is k-definite iff $S_0 = S \supset S_1 \supset S_2 \supset \ldots \supset S_k$, and $|S_k| = 1$.

Corollary 6.2. The transition system (S , δ) is k-definite iff (S_n , δ_n) is $(k - n)$-definite, for any $n = 0 , 1 , \ldots , k$.

Corollary 6.3. If the transition system (S , δ) is definite, then it is weakly $(|S| - 1)$-definite.

Corollaries 6.1. and 6.3. constitute the basis for the method of testing definiteness of a reduced finite automaton by PERLES *et al.* In § 3 we use the contractions (S_n , δ_n) to find a definite expression for the language represented by a reduced definite automaton.

2.4. To remove the restriction of the finite automaton being reduced we introduce now another type of equivalence relations in S and the corresponding contractions. As opposed to the relations Ω^n these will depend on the set of final states, and are thence defined only for transition systems of automata.

Definition 4. Two states s and s' of a finite automaton $\mathfrak{A} = (S , \delta , s_0 , F)$ are *n-indistinguishable,* $n = 0 , 1 , 2 , \ldots ,$ iff

$$\delta(s , p) \in F \text{ if and only if } \delta(s' , p) \in F ,$$

for every word p satisfying $L(p) \geq n$.

For every $n = 0 , 1 , \ldots$ we denote this equivalence relation and the corresponding state partition by Σ^n . These partitions resemble in many respects the partitions Ω^n .

Lemma 6. For every $n = 0 , 1 , \ldots ,$

(a) Σ^n is an S.P. partition,

(b) $\Sigma^n \leq \Sigma^{n+1}$,

(c) if $\Sigma^{n+1} = \Sigma^n$, then $\Sigma^{n+i} = \Sigma^n$, for any $i = 0 , 1 , 2 , \ldots ,$

(d) $\Omega^n \leq \Sigma^n$, and

(e) if the finite automaton is reduced, then $\Sigma^n = \Omega^n$.

Proof. The properties (a), (b) and (d) follow easily from definitions, (c) can be proven by the technique used in the proof of Theorem 2.

Suppose that $s\, \Sigma^n s'$ holds for a pair of states $s , s' \in S$. Then, for any words $p \in X^n$ and $r \in X^*$, $spr \in F$ iff $s'pr \in F$. Hence the states sp and $s'p$ are indistinguishable. Hence $sp = s'p$, for any $p \in X^n$, if $\mathfrak{A}$ is reduced. This implies together with (d) that $\Sigma^n = \Omega^n$.

From Theorem 1 and Lemma 6 we get

Theorem 7. A connected finite automaton $\mathfrak{A} = (S , \delta , s_0 , F)$ is k-definite if and only if

$$\Sigma^0 < \Sigma^1 < \ldots < \Sigma^{k-1} < \Sigma^k = I .$$

Definition 5. By the *nth indistinguishability (i.d.) contraction* (S_n' , δ_n') of the transition system of a finite automaton $\mathfrak{A} = (S , \delta , s_0 , F)$ we mean the Σ^n-image of (S , δ) .

As for the sets S_n we may suppose that $S_{n+1}' \subseteq S_n' \subseteq S$, for every n . To construct the i.d. contractions we do not know of such a simple recursive method as for the contractions (S_n , δ_n) . They must be constructed by first finding the partitions Σ^n and using the following obvious counterpart to Lemma 5.

Lemma 7. If $\delta(s , x) = s'$, and if $\bar{s}$ and $\bar{s}'$ are the representatives for s and s' in S_n' , then $\delta_n'(\bar{s} , x) = \bar{s}'$.

To find the partitions Σ^n we can use the following relations considered in [15].

Write $s\, \Lambda^n s'$, $n = 0 , 1 , \ldots ,$ for two states $s , s' \in S$, iff

$$sp \in F \text{ exactly in case } s'p \in F ,$$

for any word p of length n . These relations are equivalence relations in S and they can be used [15] to test the definiteness of an arbitrary connected finite automaton: the automaton is weakly k-definite if and only if $\Lambda^k = I$. In general the corresponding partitions Λ^n on S are not S.P. partitions and the relations $\Lambda^n \leq \Lambda^{n+1}$ do not hold. The partitions can be formed practically as follows.

1. $s\, \Lambda^0 s'$ iff $s , s' \in F$ or $s , s' \in S - F$.

2. $s\, \Lambda^{i+1} s'$ iff $sx\, \Lambda^i s'x$ for all x in X , $i = 0 , 1 , \ldots .$

In general it is not easy to find the partitions Σ^n , but when the automaton is k-definite, then $\Lambda^k = \Lambda^{k+1} = \Lambda^{k+2} = \ldots$, and the partitions Σ^n can be found as indicated in the following

Lemma 8. If the finite automaton $\mathfrak{A}$ is k-definite, then

$$\Sigma^n = \Lambda^n \cdot \Lambda^{n+1} \cdot \ldots \cdot \Lambda^k ,$$

for every $n = 0 , 1 , \ldots , k$.

§ 3. Testing definiteness and analysis of definite automata

3.1. Consider the following problem. Given a finite automaton it must be decided whether or not it is definite. If the answer is affirmative, the degree and a definite expression for the language represented must be found.

Perles, Rabin and Shamir [15] have given two methods to solve the questions about definiteness and degree. The first method uses the (strong) contractions (or alternatively a weaker type of contractions) and is applicable to reduced finite automata. Ginzburg [4] has formulated this test by means of graphs. The second method using the partitions Λ^n is applicable to any connected automaton.

We shall present two decision methods which also solve the analysis problem and give the language represented by a definite expression. Brzozowski [1] has given such a method involving the construction of the reverse (dual) automaton. Our second method makes use of this idea.

3.2. **Method I.** Let $\mathfrak{A} = (S , \delta , s_0 , F)$ be a given connected automaton with n states $s_0 , s_1 , \ldots , s_{n-1}$. To every word p we associate the state vector

$$t_p = (\delta(s_0 , p) , \delta(s_1 , p) , \ldots , \delta(s_{n-1} , p)) .$$

If all components of t_p belong to F we call t_p *purely positive,* and if they all belong to $S - F$, *purely negative.*

Lemma 9. A connected finite automaton is weakly k-definite if and only if t_p is for every word p of length k either purely positive or purely negative.

This lemma is a direct consequence of Theorem 1 (or 6) and it suggests the following simple testing and analysis method.

The transition table is successively written for words of length (0, 1,) 2, 3, . . . so that the row corresponding to each word p gives the vector t_p . If a row is purely positive or negative it is marked by the sign + or —, respectively. The process is continued until the first k is reached such that all rows corresponding to words of length k are marked. If there exists such a k , then $\mathfrak{A}$ is k-definite and the k-form

$$T(\mathfrak{A}) = Q + X^* R$$

is obtained from the constructed table by taking to Q all words p with $L(p) < k$ for which the first component of t_p belongs to F , and to R the words of length k marked by +. Corollary 2.1 guarantees that only a finite continuation is needed. For if no k less than n satisfies the above condition, then $\mathfrak{A}$ cannot be definite.

Example 1. Consider the 6-state automaton $\mathfrak{A}$ given by Table 1 (a), where $S = \{a , b , c , 1 , 2 , 3\}$, $s_0 = a$ and $F = \{a , b , c\}$. From the continued Table 1 (b) we see that $\mathfrak{A}$ is 2-definite and

$$T(\mathfrak{A}) = \lambda + x + X^* xy .$$

Table 1.

δ	a	b	c	1	2	3
x	b	2	2	2	3	2
y	1	c	1	1	c	c

(a)

λ	a	b	c	1	2	3	
x	b	2	2	2	3	2	
y	1	c	1	1	c	c	
xx	2	3	3	3	2	3	—
xy	c	c	c	c	c	c	+
yx	2	2	2	2	2	2	—
yy	1	1	1	1	1	1	—

(b)

The continuation is facilitated if we note that the column under any state s corresponding to the words xp , with $x \in X$ and $L(p) = j$ $(j = 1 , 2 , . . .)$, is the same as that under $\delta(s , x)$ corresponding to the words of length j , when the words of equal length are always written in the same alphabetical order.

If the automaton has proven to be k-definite, then the indistinguishable states can be found from the continued table, for obviously any pair of distinguishable states can be distinguished by a word of length less than k . Thus in Example 1 we find from Table 1 (b) considering t_λ , t_x and t_y that the states 2 and 3 are indistinguishable.

The procedure described above can often be considerably shortened and the canonical expression (cf. [1]) is obtained by omitting in the continuation every word having as a suffix some word previously marked. Then for k-definiteness it is necessary and sufficient that the table has some rows for words of length k, which are all marked and thus no rows for words of length more than k. The canonical representation

$$T(\mathfrak{A}) = Q + X^* R$$

is obtained by taking to R, without changing their order, all words marked by $+$ and to Q all other words whose state vector has a final state as the first component. Also from this table the indistinguishable states can be found.

Example 2. In the case indicated by Table 2, where $s_0 = 1$ and $F = \{a, b\}$, the table is continued for only those words of length 3 which end with xy and only for those words of length 4 which end with xxy. The language represented is 4-definite and we get the canonical expression

$$T(\mathfrak{A}) = x + xxy + X^* yy .$$

Table 2.

λ	1	2	3	a	b	
x	a	3	3	2	3	
y	2	b	2	2	b	
xx	2	3	3	3	3	—
xy	2	2	2	b	2	
yx	3	3	3	3	3	—
yy	b	b	b	b	b	+
xxy	b	2	2	2	2	
yxy	2	2	2	2	2	—
$xxxy$	2	2	2	2	2	—
$yxxy$	2	2	2	2	2	—

Instead of the 31 rows of the complete table Table 2 has only 11 rows.

Method I is technically simple and it gives the languages represented by any choice of initial state, but the maximal length of the table increases rapidly with the number of states.

3.3. **Method II.** This method involves two stages. First sufficiently many contractions of the transition system of $\mathfrak{A}$ are formed and from these the definiteness is tested. If the automaton proves to be definite, then the language represented is obtained as a definite expression from a graph which is constructed by means of the contractions.

First let $\mathfrak{A}$ be reduced. The contractions (S_n , δ_n) are successively formed as indicated in 2.3. The initial state is always taken as the representative for its class in any contraction. Then Theorem 3 and Corollary 6.1. provide the criterion of PERLES *et al.*: if (S_k , δ_k) is the first contraction admitting no further proper contractions, then $\mathfrak{A}$ is definite iff $|S_k| = 1$. If $|S_k| = 1$, then $\deg T(\mathfrak{A}) = k$.

Suppose that $\mathfrak{A}$ has proven to be k-definite. Then construct a graph $\Gamma(\mathfrak{A})$ as follows.

The nodes of $\Gamma(\mathfrak{A})$ correspond to subsets of S and we assign them orders in the following way.

1. The initial node $s_\lambda^- = F$ is the only node of order 0.

2. When the nodes of order $i \; (i = 0 , 1 , \ldots)$ are constructed, then the nodes of order $i + 1$ are obtained from the table of (S_i , δ_i) as follows. To each node $s_p^- \neq S_{i-1}$ (for $i = 0 , s_p^- \neq S$) of order i and letter x we introduce a node

$$s_{px}^- = \{s \in S_i | \delta_i(s , x) \in s_p^-\} ,$$

if this is nonempty, and lead a transition labelled by x from s_p^- to s_{px}^-. Equal nodes of the same order may be identified as one node. If $s_{px}^- = \emptyset$, then a transition labelled by x is led to a common empty node $s_\emptyset^-$ (In practice this node and transitions leading to it may be neglected).

3. If for a node s_p^- of order i , $s_p^- = S_{i-1}$ (for $i = 0$, $s_p^- = S$) , i.e., if s_p^- contains all states of the transition system from which it was constructed, then no nodes s_{px}^- are introduced, but a transition labelled by all input letters is led from s_p^- to itself.

Evidently the construction terminates and the last nonempty nodes are of order k or $k - 1$.

Let δ^- be the transition function of $\Gamma(\mathfrak{A})$ and denote by $\mathfrak{A}_s$ the finite automaton (S , δ , s , F) for any state s in S . If we further denote by p^- the reverse of the word p (i.e., $p^- = x_m x_{m-1} \ldots x_1$ if $p = x_1 x_2 \ldots x_m$, and $\lambda^- = \lambda$) , then the connection between $\Gamma(\mathfrak{A})$ and $\mathfrak{A}$ can be explained by the following

Theorem 8. Let p be an arbitrary word. Then for any state s in S: $p^- \in T(\mathfrak{A}_s)$ if and only if $\delta^-(s_\lambda^- , p)$ contains a state identified with s in the contraction from which $\delta^-(s_\lambda^- , p)$ was constructed.

Proof. First suppose that $\delta^-(s_\lambda^- , p)$ contains a state $\bar{s}$ identified with s . From the construction of $\Gamma(\mathfrak{A})$ it follows that the order of

$\delta^-(s_\lambda^-, p)$ does not exceed the length m of the word p. Hence $s\,\Omega^m\bar{s}$ by Theorem 6. On the other hand it follows from the construction of the graph that $\delta(\bar{s}, p^-) \in s_\lambda^- = F$. Hence also $\delta(s, p^-)$ belongs to F.

Suppose now that $p^- \in T(\mathfrak{A}_s)$. We proceed by induction on $L(p) = m$. If $m = 0$, then $p = \lambda$ and $s \in F = s_\lambda^-$. Suppose that the theorem is true for any state and any word of length i, and let m be $i + 1$. We write $p = qx$. Denote $\bar{s} = C^j s$ and $s' = C^j sx$, where $j \leq i$ is the order of $\delta^-(s_\lambda^-, q)$. Then by Lemma 5 $\delta_j(\bar{s}, x) = s'$. Hence

$$\bar{s} \in \{s'' \in S_j | \ \delta_j(s'', x) \in \delta^- (s_\lambda^-, q)\} = \delta^-(s_\lambda^-, p)\,.$$

The definite expression $T(\mathfrak{A}) = Q + X^* R$ can be found from $\Gamma(\mathfrak{A})$ by taking to R the reverses of all words of length i leading from the initial node to a node S_{i-1} of order i $(i = 1, \ldots, k)$ and to Q the reverses of all words leading to some other node containing the initial state s_0 of $\mathfrak{A}$.

Example 3. Consider the finite automaton $\mathfrak{A} = (S, \delta, s_0, F)$ with $F = \{a, b, c\}$ and $s_0 = 1$ given by Table 3. The contractions (S_1, δ_1) and (S_2, δ_2) are given in the same table. From (S_2, δ_2) we see that $\mathfrak{A}$ is 3-definite because $|S_3| = 1$. The graph $\Gamma(\mathfrak{A})$ is shown in Figure 1. The nodes $s_\lambda^- = \{a, b, c\}$, $s_x^- = \{1\}$ and $s_y^- = \{3, 4, a, c\}$ were read from (S_0, δ_0), $s_{yx}^- = S_1$ and $s_{yy}^- = \{2, 4\}$ from (S_1, δ_1), and $s_{yyy}^- = S_2$ from (S_2, δ_2). The empty node and transitions leading to it have been omitted. From the graph we get

$$T(\mathfrak{A}) = x + X^*(xy + yyy)\,.$$

Table 3.

δ_0	1	2	3	4	a	b	c	1-eq. states
x	a	3	3	3	3	3	3	$\{2, b\}$, $\{3, a\}$,
y	2	4	b	c	b	4	c	$\{4, c\}$
δ_1	1	2	3	4				
x	3	3	3	3				$\{1, 3\}$, $\{2, 4\}$
y	2	4	2	4				
δ_2	1	2						
x	1	1						$\{1, 2\}$
y	2	2						

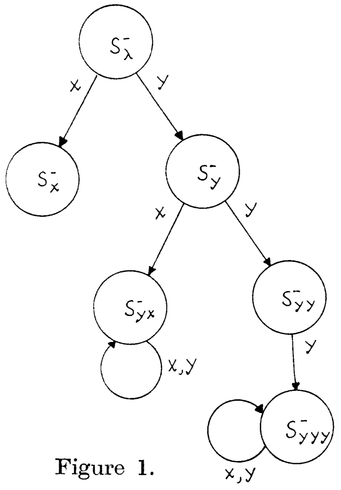

Figure 1.

If the finite automaton $\mathfrak{A}$ is not known to be reduced, then the contractions must be replaced by the i.d. contractions. The analysis procedure consists then of three stages.

1. The partitions Λ^n $(n = 0 , 1 , 2 , \ldots)$ are computed by the recursive formulas of PERLES *et al.* (see § 2) until a partition $\Lambda^k = I$ is obtained. The possible nondefiniteness is revealed at this stage. For if no $k < |S|$ satisfies this condition, then $\mathfrak{A}$ cannot be definite.

2. If the automaton has proven to be k-definite, the partitions Σ^n for $n = k , k - 1 , \ldots , 0$ are computed using Lemma 8, and the corresponding i.d. contractions are formed using Lemma 7.

3. A graph $\Gamma'(\mathfrak{A})$ is constructed just as $\Gamma(\mathfrak{A})$ but using instead of the contractions (S_n , δ_n) the i.d. contractions (S_n' , δ_n').

A definite expression for $T(\mathfrak{A})$ can be read from $\Gamma'(\mathfrak{A})$ in the same way as from $\Gamma(\mathfrak{A})$ above. The justification of this method can be given along the same lines as in the method for a reduced automaton.

§ 4. Definite automaton partitions and synthesis of definite automata

4.1. A partition Π on X^* is called an *automaton partition* (or *right invariant*) iff for each class $\Pi_\alpha \in \Pi$ and letter $x \in X$ there exists a block $\Pi_\beta \in \Pi$ such that $\Pi_\alpha x \subseteq \Pi_\beta$.

GLUSHKOV [5], [6] has shown that the automaton partitions form a lattice relative to the partial ordering $\leq$. The infimum and supremum of the automaton partitions Π^1 and Π^2 are given by $\Pi^1 \cdot \Pi^2$ and $\Pi^1 + \Pi^2$, respectively.

An automaton partition Π defines an initial transition system

$$A(\Pi) = (\Pi , \delta , \Pi_\lambda) ,$$

where $\Pi_\lambda \in \Pi$ is the class containing λ, and $\delta : \Pi \times X \to \Pi$ is defined by the condition

$$\delta(\Pi_\alpha , x) = \Pi_\beta \text{ iff } \Pi_\alpha x \subseteq \Pi_\beta ,$$

for any Π_α, $\Pi_\beta \in \Pi$ and $x \in X$.

Conversely, let $A = (S , \delta , s_0)$ be a connected initial transition system and π a partition on S. By the input partition defined by π we mean the partition

$$\Pi\{\pi\} = \{T(\mathfrak{A}_\alpha) \mid \pi_\alpha \in \pi\} ,$$

where $\mathfrak{A}_\alpha = (S, \delta, s_0, \pi_\alpha)$. The following theorem is a generalization of a result implicit in the well-known representability theorem by NERODE (cf. [16]) and explicitly presented in [6].

Theorem 9. Let $A = (S, \delta, s_0)$ be a connected initial transition system and π a partition on S. Then $\Pi = \Pi\{\pi\}$ is an automaton partition if and only if π is an S.P. partition.

Proof. Let p_1 and p_2 be two words in the same class $\Pi_\alpha \in \Pi$, and x a letter. Then $s_0 p_1 \pi s_0 p_2$, and if π is an S.P. partition also $s_0 p_1 x \pi s_0 p_2 x$, which implies $p_1 x \Pi p_2 x$.

On the other hand, if π is not an S.P. partition, then there exist classes $\pi_\alpha, \pi_\beta \in \pi$, states $s_1, s_2 \in \pi_\alpha$, and a letter x such that $s_1 x \in \pi_\beta$ but $s_2 x \notin \pi_\beta$. Because A is connected there exist words p_1 and p_2 such that $s_1 = s_0 p_1$ and $s_2 = s_0 p_2$. Now $p_1 \Pi p_2$, but not $p_1 x \Pi p_2 x$.

We note that all automaton partitions defined by S.P. partitions of finite transition systems are finite. The following theorem reveals the isomorphism between the lattice of the S.P. partitions of a given initial transition system and a sublattice of the automaton partitions.

Theorem 10. Let $A = (S, \delta, s_0)$ be a connected initial transition system, and π^1 and π^2 two of its S.P. partitions. Then

(a) $\pi^1 < \pi^2$ implies $\Pi\{\pi^1\} < \Pi\{\pi^2\}$,

(b) $\Pi\{\pi^1 \cdot \pi^2\} = \Pi\{\pi^1\} \cdot \Pi\{\pi^2\}$, and

(c) $\Pi\{\pi^1 + \pi^2\} = \Pi\{\pi^1\} + \Pi\{\pi^2\}$.

The proof of this theorem is staightforward.

4.2. We call an input partition *weakly k-definite* iff all of its classes are weakly k-definite languages. The notions of *k-definite* and *definite partitions,* and their *degree* are introduced as the corresponding concepts for languages in Definition 1. Hence a partition is k-definite iff all of its classes are weakly k-definite languages and at least one is k-definite. We note that a definite input partition is always finite.

Theorem 11. The infimum and supremum of two weakly k-definite input partitions $(k = 0, 1, \ldots)$ are weakly k-definite. Hence the definite input partitions form a lattice relative to the relation $\leq$, and the weakly k-definite $(k = 0, 1, \ldots)$ partitions a sublattice of this lattice.

The easy proof based on the fact that the unions and intersections of weakly k-definite languages are weakly k-definite is omitted. As a consequence we get

Corollary 11.1. The definite (weakly k-definite) automaton partitions form a lattice relative to $\leq$.

Theorem 12. Let $\mathfrak{A} = (S , \delta , s_0 , F)$ be a k-definite connected finite automaton and Σ^n $(n = 0 , 1 , \ldots , k)$ the state partition into n-indistinguishable states. Then $\Pi^n = \Pi\{\Sigma^n\}$ is a $(k - n)$-definite automaton partition.

Proof. As noted in Lemma 6 Σ^n is an S.P. partition and Π^n thus by Theorem 9 an automaton partition.

Let $p \in X^{k-n}$ and $q \in X^*$ be chosen arbitrarily. For any word r satisfying $L(r) \geq n$, we have $L(pr) \geq k$, and thus $s_0pr \in F$ iff $s_0qpr \in F$. Hence $s_0p\, \Sigma^n s_0qp$ and $p\Pi^n qp$. This implies that Π^n is weakly $(k - n)$-definite.

By Theorem 7 $\Sigma^k = I$ and thus Π^k is 0-definite. Suppose that $k > 0$ and $0 \leq n \leq k - 1$. Because $\mathfrak{A}$ is not weakly $(k - 1)$-definite there exist two words $p \in X^{k-1}$ and $q \in X^*$ such that $s_0p \in F$ but $s_0qp \notin F$, or $s_0p \notin F$ but $s_0qp \in F$. Denote $p = p'p''$, where $L(p') = k - n - 1$ and $L(p'') = n$. Then p' and qp' belong to distinct classes in Π^n because $s_0p'\, \Sigma^n s_0qp'$ does not hold. Thus Π^n is not weakly $(k - n - 1)$-definite.

We get the following corollaries with the assumptions of Theorem 12.

Corollary 12.1. The state partition of $\mathfrak{A}$ into indistinguishable states defines a k-definite automaton partition.

Corollary 12.2. If $\Sigma^n \leq \pi$, for the state partition π, then $\Pi\{\pi\}$ is weakly $(k - n)$-definite.

Corollary 12.3. If $\mathfrak{A}$ is reduced, then each of its state partitions defines a weakly k-definite input partition.

4.3. Let C be a family of partitions on X^*. The automaton partition Π is a *maximal automaton refinement* of C iff

(1) Π refines every partition in C, and

(2) Π is not a proper refinement of any automaton partition which refines every partition in C.

GLUSHKOV [6] has shown that each family C has a unique maximal automaton refinement.

Let $K = \{U_1 , U_2 , \ldots , U_n\}$ be a nonempty collection of languages. Each U_i defines an input partition $\psi^i = \{U_i , X^* - U_i\}$. When all languages in K are regular, then the maximal automaton refinement

Π of $\{\psi^1, \psi^2, \ldots, \psi^n\}$ is finite and all languages of K are representable in the initial transition system $A(\Pi)$. Moreover it was shown in [6] that every initial transition system in which the languages of K are representable can be mapped homomorphically onto $A(\Pi)$. Hence $A(\Pi)$ is minimal. Letitsevski [10] has given the following method for finding Π.

First define Π^0 as the input partition $\Pi^0 = \psi^1 \cdot \psi^2 \cdot \ldots \cdot \psi^n$. Suppose that Π^i has been constructed $(i = 0, 1, \ldots)$. Then define a refinement Π^{i+1} of Π^i dividing each block $\Pi^i_\alpha \in \Pi^i$ in maximal subsets $\Pi^i_{\alpha\beta} = \Pi^{i+1}_\gamma$ as follows. For any letter x there exists a class $\Pi^i_\varepsilon \in \Pi^i$ such that $\Pi^{i+1}_\gamma x \subseteq \Pi^i_\varepsilon$. In this way a descending sequence

$$\Pi^0 > \Pi^1 > \Pi^2 > \ldots$$

of input partitions (termed here Letitsevski partitions) is obtained. The procedure is stopped when a partition Π^m is reached for which no refinement is needed, i.e., $\Pi^{m+1} = \Pi^m$. This Π^m is the maximal automaton refinement, and each U_i of K is represented in $A(\Pi^m)$ by the set of states corresponding to the classes constituting U_i. We do not know any practical method for finding the refinements Π^i in the case where the languages in K are general regular languages. In Section 4.4. we present such a method for definite languages.

Lemma 10. Let Π^0, Π^1, $\Pi^2, \ldots$ be the Letitsevski partitions for some collection K of languages, and i, j a pair of non-negative integers satisfying $i \geq j$. For any word p of length $i - j$ and any class $\Pi^i_\alpha \in \Pi^i$ there exists a class $\Pi^j_\beta \in \Pi^j$ such that $\Pi^i_\alpha p \subseteq \Pi^j_\beta$.

The lemma is easily established by induction on $i - j$. In the following theorem we give an upper bound for the number of refinements needed when all languages in the family are definite.

Theorem 13. Let $K = \{U_1, U_2, \ldots, U_n\}$ $(n \geq 1)$ be a collection of weakly k-definite languages and $\Pi^0, \Pi^1, \ldots$ the corresponding Letitsevski partitions. If $k > 0$, then Π^{k-1} is an automaton partition. If $k = 0$, then Π^0 is an automaton partition.

Proof. The cases $k = 0$ and $k = 1$ are trivial. We suppose now that $k > 1$. Let Π^{k-1}_α be an arbitrary class in Π^{k-1}. We prove by reversed induction on $h = k - 1, \ldots, 1$ that for every word p of length h there exists a class $\Pi^{k-h}_\beta \in \Pi^{k-h}$ such that $\Pi^{k-1}_\alpha p \subseteq \Pi^{k-h}_\beta$.

First we note that for every word p, with $L(p) \geq k$, there exists a class $\Pi^0_\gamma \in \Pi^0$ such that $\Pi^{k-1}_\alpha p \subseteq \Pi^0_\gamma$. This follows from the weakly k-definiteness of the languages in K which implies that the class of a word qp in Π^0 is uniquely determined by p.

Suppose now that $L(p) = k - 1$. By Lemma 10 there exists a class $\Pi^0_\beta \in \Pi^0$ such that $\Pi^{k-1}_\alpha p \subseteq \Pi^0_\beta$. For any x in X, $L(px) = k$ and thus there exists a class $\Pi^0_\gamma \in \Pi^0$ such that $\Pi^{k-1}_\alpha px \subseteq \Pi^0_\gamma$. But this implies that all words of $\Pi^{k-1}_\alpha p$ must belong to the same class $\Pi^1_\beta \in \Pi^1$ because they belong to the same class Π^0_β in Π^0. The first step of the induction is thus established.

Suppose now that the proposition is true for some $h = k - 1, \ldots, 2$. Let p be a word of length $h - 1$. By the induction hypothesis there exists for every letter x a class $\Pi^{k-h}_\beta \in \Pi^{k-h}$ such that $\Pi^{k-1}_\alpha px \subseteq \Pi^{k-h}_\beta$. On the other hand $\Pi^{k-1}_\alpha p \subseteq \Pi^{k-h}_\gamma$ for some $\Pi^{k-h}_\gamma \in \Pi^{k-h}$ by Lemma 10. Thus the words of $\Pi^{k-1}_\alpha p$ belong to the same class in Π^{k-h+1}. This completes the induction, and for $h = 1$ we get the result that, for any letter x there exists a class $\Pi^{k-1}_\beta \in \Pi^{k-1}$ such that $\Pi^{k-1}_\alpha x \subseteq \Pi^{k-1}_\beta$.

4.4. We now describe a method based on the above considerations to find the minimal initial transition system in which all languages of a collection of definite languages are representable.

Let $C = \{U_1, U_2, \ldots, U_n\}$ $(n \geq 1)$ be a collection of definite languages and k the least integer such that all languages in C are weakly k-definite. Write a table having a row for each word of length k or less. To each language associate a column such that the entry corresponding to a word p contains 1 (0) if $p \in U_i$ $(p \notin U_i)$. Hence each word in the table will be indexed by a sequence of length n of 1's and 0's. If $n > 1$, these sequences are replaced by simple indexes (e.g. by taking the sequences as binary numbers). After this index column write a column for each input letter so that in the column of x in the row of p we write the index of px, or if $L(p) = k$ the index of $px|k$. Hereby every word gets a new sequence of indexes. These are again replaced by simple indexes so that two words get the same index iff their index sequences are identical. This procedure is repeated until every pair of words having equal indexes also gets equal index rows. These last indexes can be taken as the states of the transition system to be constructed, and the last columns give the corresponding transition table (vertically written and with eventual repetitions). The initial state is the index of λ and the final states representing the language U_i are the indexes of the words in the table which belong to U_i. Theorem 13 guarantees that at most k (or one if $k = 0$) indexing steps are needed. Observation of the obvious periodicities in the formation of the table facilitates the work considerably.

Example 4. We construct the minimal transition system in which the 2-definite languages $U_1 = \lambda + x + X^* (xx + xy)$ and $U_2 = x + y + X^* (xx + yx + yy)$ are representable . The construction is given by Table 4 and the obtained transition system by Table 5. The initial state is a , and U_1 is represented by $\{a , b , d\}$ and U_2 by $\{b , c , e\}$.

Table 4.

	U_1	U_2	xy	xy
λ	1	0	$= 231$	$= abc$
x	1	1	$= 332$	$= bbd$
y	0	1	$= 111$	$= cec$
xx	1	1	$= 332$	$= bbd$
xy	1	0	$= 211$	$= dec$
yx	0	1	$= 132$	$= ebd$
yy	0	1	$= 111$	$= cec$

Table 5.

	a	b	c	d	e
x	b	b	e	e	b
y	c	d	c	c	d

4.5. Let p be any word of length k . From the construction in 4.4 we see that $X^* p$ is a subset of some class Π_β of the maximal automaton refinement Π . On the other hand, for any class $\Pi_\alpha \in \Pi$, $\Pi_\alpha p \subseteq X^* p$. Thus in $A(\Pi)$ $\delta(\Pi_\alpha , p) = \Pi_\beta$ for any state $\Pi_\alpha \in \Pi$. Hence we get the following generalization of Theorem 3.

Theorem 14. The minimal initial transition system in which a given collection of weakly k-definite languages is representable is weakly k-definite.

§ 5. Definite sequential machines

5.1. A *Mealy machine* is an ordered quadruple $\mathfrak{M} = (S , X , \delta , \omega)$, where S (state set), X (input alphabet) and Y (output alphabet) are finite nonempty sets,

$$\delta\colon\ S \times X \rightarrow S \text{ (next-state function)}$$

and

$$\omega\colon\ S \times X \rightarrow Y \text{ (output function)}$$

functions. The domains of δ and ω are extended to $S \times X^*$ in the usual way. If ω can be considered as a function from S to Y , then $\mathfrak{M}$ is a *Moore machine.*

Adding to the quadruple a specified initial state s_0 we get an initial Mealy machine. The language represented by the set Y_1 of output letters in the initial Mealy machine is denoted by $T(\mathfrak{M}, s_0, Y_1)$, when the initial state is s_0.

Clearly, the only 0-definite language representable in a Mealy-machine is $\emptyset$. Excluding this trivial case we restrict the consideration to k-definite languages with $k > 0$.

5.2. We consider now the transition system (S, δ) of a Mealy machine $\mathfrak{M}$ for which $T(\mathfrak{M}, s_0, Y_1)$ is definite, for some s_0 and Y_1. That $\mathfrak{M}$ is reduced as a sequential machine does not imply the definiteness of its transition system. We define a stronger concept of reducedness.

Two states $s, s' \in S$ are said to be *Y_1-distinguishable,* for some $Y_1 \subseteq Y$, iff there exists a word p such that $\omega(s, p) \in Y_1$ but $\omega(s', p) \notin Y_1$, or vice versa. We call $\mathfrak{M}$ *Y_1-reduced* iff every pair of its states is Y_1-distinguishable.

Theorem 15. Let $\mathfrak{M}$ be a Mealy machine connected for the initial state s_0. If there exists a set Y_1 of output letters such that $T(\mathfrak{M}, s_0, Y_1)$ is k-definite $(k \geq 1)$ and $\mathfrak{M}$ is Y_1-reduced, then $\mathfrak{M}$ has a $(k-1)$-definite transition system.

Conversely, if a Mealy machine $\mathfrak{M}$ has a k-definite $(k \geq 0)$ transition system, then $T(\mathfrak{M}, s_0, Y_1)$ is weakly $(k+1)$-definite for any choice of $s_0 \in S$ and $Y_1 \subseteq Y$.

Proof. Suppose that the conditions of the first part are fulfilled, but that there exists a word p of length $k-1$ and states s_1, s_2 such that $s_1' = \delta(s_1, p) \neq \delta(s_2, p) = s_2'$. Let p_1 and p_2 be words such that $s_1 = \delta(s_0, p_1)$ and $s_2 = \delta(s_0, p_2)$. For any word q of length greater than or equal to 1 $L(pq) \geq k$. Thus

$$\omega(s_0, p_1pq) \in Y_1 \text{ iff } \omega(s_0, p_2pq) \in Y_1,$$

or equivalently,

$$\omega(s_1', q) \in Y_1 \text{ iff } \omega(s_2', q) \in Y_1.$$

This contradicts the assumption that $\mathfrak{M}$ is Y_1-reduced. Hence the transition system must be weakly $(k-1)$-definite. That it is $(k-1)$-definite follows from the converse part of the theorem. This, in turn, follows from the fact that, for any $s_0 \in S$ and $Y_1 \subseteq Y$,

$$T(\mathfrak{M}, s_0, Y_1) = \bigcup_{x \in X} T_x x,$$

where for each $x \in X$, T_x is the weakly k-definite language represented in the initial transition system (S , δ , s_0) by the set

$$S_x = \{s \in S | \, \omega(s , x) \in Y_1\}$$

of final states.

Clearly the minimal Mealy machine representing a k-definite $(k \geq 1)$ language satisfies the conditions of the first part of Theorem 15. Thus we get the following corollaries.

Corollary 15.1. The minimal Mealy machine representing a k-definite $(k \geq 1)$ language has a $(k - 1)$-definite transition system.

Corollary 15.2. A Mealy machine representing a k-definite $(k \geq 1)$ language has at least k states.

From Theorem 14 it follows that the minimal Moore machine representing a set of weakly k-definite languages has a weakly k-definite transition system. As shown by Ibarra [8] the equivalent minimal Mealy machine can be constructed from this by identifying all 1-equivalent states. Thus the transition system of the minimal Mealy machine is the 1*st* contraction of the transition system of the Moore machine. Using Corollaries 6.2. and 6.1. we get

Theorem 16. The minimal Mealy machine representing a collection of weakly k-definite $(k \geq 1)$ languages has a $(k - 1)$-definite transition system.

If the corresponding minimal Moore machine has n states, then the minimal Mealy machine has at most $n - 1$ states.

5.3. We now modify the synthesis and analysis methods presented above for Mealy machines.

The minimal Mealy machine representing a collection of definite languages is obtained by assigning outputs to the states of the minimal initial transition system constructed by the partition method and transforming this Moore machine into a Mealy machine by the method of Ibarra.

Let $\mathfrak{M} = (S , X , Y , \delta , \omega)$ be connected for the initial state s_0 , and $Y_1 \subseteq Y$ a given set of representing output letters. To test the definiteness of $T(\mathfrak{M} , s_0 , Y_1)$ and to find, in the affirmative case, a definite regular expression for it, we construct a graph $\Gamma(\mathfrak{M} , Y_1)$ as follows.

1. The initial node is Y_1 .

2. For each $x \in X$ introduce a node corresponding to the set S_x ,

and lead a transition from Y_1 to S_x labelled by x. Nodes corresponding to equal sets S_x are identified, and nodes corresponding to an empty set S_x may be omitted.

3. Starting from each nonempty S_x the graph $\Gamma(\mathfrak{A}_x)$ is constructed as in § 3 for the finite automaton $\mathfrak{A}_x = (S, \delta, s_0, S_x)$. If we know that $\mathfrak{M}$ is Y_1-reduced, then the contractions (S_n, δ_n) are first constructed and the subgraphs $\Gamma(\mathfrak{A}_x)$ can be found by means of them.

Clearly $T(\mathfrak{M}, s_0, Y_1)$ is definite iff all languages $T(\mathfrak{A}_x)$ are definite. More accurately — if the languages $T(\mathfrak{A}_x)$ are all definite and at least one of them nonempty, then

$$\deg T(\mathfrak{M}, s_0, Y_1) = \max_{x \in X} T(\mathfrak{A}_x) + 1 .$$

A definite expression for $T(\mathfrak{M}, s_0, Y_1)$ can be found from $\Gamma(\mathfrak{M}, Y_1)$ just as $T(\mathfrak{A})$ was found from $\Gamma(\mathfrak{A})$ in § 3.

§ 6. Compositions of definite transition systems

6.1. We consider now the preservation of definiteness in various compositions of transition systems.

In the following theorem the *parallel connection* may be understood in any of the current meanings of the word. The serial connection is defined in the following general way (cf. [6]).

Let $A_1 = (S_1, \delta_1)$ and $A_2 = (S_2, \delta_2)$ be transition systems over the alphabets X_1 and X_2, respectively, and let φ: $S_1 \times X_1 \rightarrow X_2$ be any mapping. Then the transition system $A_\varphi = (S_1 \times S_2, \delta)$ over the alphabet X_1 is a *serial connection* of A_1 and A_2, where δ is defined by the condition

$$\delta((s_1, s_2), x) = (\delta_1(s_1, x), \delta_2(s_2, \varphi(s_1, x))),$$

for any $(s_1, s_2) \in S_1 \times S_2$ and $x \in X_1$.

Theorem 17. The parallel connection A of any definite transition systems $A_1, \ldots, A_n$ is definite, and

$$\deg A = \max\{\deg A_1, \ldots, \deg A_n\} .$$

Let $A_1 = (S_1, \delta_1)$ and $A_2 = (S_2, \delta_2)$ be definite transition systems over the alphabets X_1 and X_2, respectively. Their series connection A_φ is definite and

$$\deg A_\varphi \leq \deg A_1 + \deg A_2 ,$$

for any mapping φ: $S_1 \times X_1 \rightarrow X_2$.

Proof. The first part of the theorem is obvious. We prove the second proposition.

The mapping φ can be extended to a mapping φ: $S_1 \times X_1^* \rightarrow X_2^*$ putting for any $s \in S_1$ $\varphi(s , \lambda) = \lambda$ and

$$\varphi(s , p) = \varphi(s , x_1) \varphi(sx_1 , x_2) \ldots \varphi(sx_1 \ldots x_{n-1} , x_n) ,$$

for any word $p = x_1x_2 \ldots x_n$ $(n \geq 1)$. Then for any $(s_1 , s_2) \in S_1 \times S_2$ and $p \in X_1^*$

$$\delta((s_1 , s_2) , p) = (\delta_1 (s_1 , p) , \delta_2(s_2 , \varphi(s_1 , p))) .$$

Let (s_1 , s_2) and (s_1' , s_2') be arbitrary states of A_φ and $p = p_1p_2$ any word of length $k_1 + k_2$, where $L(p_1) = k_1 = \deg A_1$ and $L(p_2) = k_2 = \deg A_2$. Now $\delta_1(s_1 , p_1) = \delta_1(s_1' , p_1)$ because A_1 was k_1-definite. Hence

$$\varphi(s_1 , p) = \varphi(s_1 , p_1) \varphi(s_1p_1 , p_2)$$

and

$$\varphi(s_1' , p) = \varphi(s_1' , p_1) \varphi(s_1'p_1 , p_2)$$

have a common k_2-suffix, and thence

$$\delta_2(s_2 , \varphi(s_1 , p)) = \delta_2(s_2' , \varphi(s_1' , p)) .$$

Thus we get

$$\begin{aligned} \delta((s_1 , s_2) , p) &= (\delta_1(s_1 , p) , \delta_2(s_2 , \varphi(s_1 , p))) \\ &= (\delta_1(s_1' , p) , \delta_2(s_2' , \varphi(s_1' , p))) \\ &= \delta((s_1' , s_2') , p) . \end{aligned}$$

Hence A_φ is weakly $(k_1 + k_2)$-definite. The actual degree of A_φ depends on φ .

Corollary 17.1. Any series-parallel connection of definite transition systems is definite.

Corollary 17.2. A language representable in a series-parallel connection of finite automata or sequential machines with definite transition systems is definite.

Of course, definiteness is not retained in general in a feedback connection of definite transition systems.

CHAPTER II

NONINITIAL AUTOMATA

§ 7. Representability in noninitial finite automata

7.1. A *noninitial automaton* over the alphabet X is an ordered triple (S, δ, F), where (S, δ) is a transition system and $F \subseteq S$ a set of final states. The noninitial automata considered here are always finite, i.e. S is a finite set.

Consider the following straightforward extension of the concept of representation from finite automata to noninitial automata; The language U is represented in $\mathfrak{A} = (S, \delta, F)$ iff $p \in U$ implies $\delta(s, p) \in F$ for any $s \in S$, and $p \notin U$ implies $\delta(s, p) \notin F$, for any $s \in S$. It is easy to see that then only the languages $\emptyset$ and X^* are representable in a noninitial automaton. This was noted by STARKE [19] and he introduced a weaker type of representation for noninitial automata. Our definition differs from that of STARKE in that the restriction is imposed only on the mode of representation and not on the words of the language represented.

Definition 6. The noninitial automaton $\mathfrak{A} = (S, \delta, F)$ *k-represents* (k a non-negative integer) a language U iff, for any word p satisfying the condition $L(p) \geq k$,

(i) $\delta(s, p) \in F$, for all states $s \in S$, if $p \in U$, and
(ii) $\delta(s, p) \notin F$, for all states $s \in S$, if $p \notin U$.

Hence the noninitial automaton is required to decide only about words possessing a given minimum length k whether or not they belong to U.

The following results formulated in our terminology can be derived from [19].

A language k-represented in a noninitial automaton is weakly k-definite, and for any weakly k-definite language (in [19] of type $X^* R$, $R \subseteq X^k$) a noninitial automaton representing it can be found.

7.2. Obviously the conditions for k-representation are still so strict that it may happen that a given noninitial automaton does not k-represent any language for any k. We consider the analysis problem of deciding whether a noninitial automaton k-represents some language for some k and finding the languages k-represented.

Theorem 18. The noninitial automaton $\mathfrak{A} = (S, \delta, F)$ k-represents $(k \geq 0)$ some language if and only if

(a) all languages $T_i = T(\mathfrak{A}_i)$, where $\mathfrak{A}_i = (S, \delta, s_i, F)$ and $s_i \in S$, are weakly k-definite and almost equal.

If (a) is satisfied, then $\mathfrak{A}$ k-represents a language T if and only if T is weakly k-definite and almost equal to the languages T_i.

Proof. Suppose that the language T is k-represented in $\mathfrak{A}$. Consider any state $s_i \in S$ and let p be an arbitrary word with $L(p) \geq k$. Write $p = qr$, where $r = p|k$. If $p \in T_i$, then $s_i p \in F$ and hence $p \in T$. Furthermore $(s_i q) r \in F$ implies that $r \in T$, and hence $s_i r \in F$, which implies $r \in T_i$. Similarly we see that $r \in T_i$ implies $r \in T$. Hence especially $(s_i q) r = s_i p \in F$, and so $p \in T_i$ and $p \in T$. Therefore T_i is weakly k-definite and any of its words of length k or more belongs to T.

On the other hand, if $p \in T$ and $L(p) \geq k$, then $s_i p \in F$ and thus $p \in T_i$. Hence T and T_i are almost equal, for they can differ only by words of length less than k. From this we can conclude that T is weakly k-definite, and that all languages T_i are almost equal.

Conversely, assume that the languages T_i are weakly k-definite and almost equal. Then they have k-forms $T_i = Q_i + X^* R$, where $R \subseteq X^k$ is the same set for every i. Let T be any weakly k-definite language almost equal to the languages T_i. Then T has the k-form $Q + X^* R$. Hence for any state s_i and any word p satisfying $L(p) \geq k$:

$$s_i p \in F \text{ iff } p \in T_i \text{ iff } p \in X^* R \text{ iff } p \in T .$$

Thus T is k-represented.

Consider now the state vectors t_p used in Section 3.2. Either by Theorem 18 or directly from definitions we get

Corollary 18.1. The noninitial automaton $\mathfrak{A}$ k-represents some language if and only if the vector t_p is either purely positive or purely negative, for any word p of length k. If this condition is satisfied, then $\mathfrak{A}$ k-represents the weakly k-definite languages almost equal to $X^* R$, where $R = \{p \in X^k | \; t_p \text{ purely positive}\}$.

Hence the analysis of a noninitial automaton can be accomplished by Method I described in Section 3.2. Note that the transition system of $\mathfrak{A}$ need now not be connected for any initial state.

If $\mathfrak{A}$ is connected for some initial state — say s_0, then for any state s_i there exists a word p such, that $s_0 p = s_i$. Hence $T_i = \partial_p T_0$ is by Lemma 2 weakly k-definite and almost equal to T_0 whenever T_0 is weakly k-definite. Thus we get

Corollary 18.2. If the noninitial automaton $\mathfrak{A}$ is connected for some initial state s_0, then $\mathfrak{A}$ k-represents some language if and only if the corresponding initial automaton $\mathfrak{A}_0$ is weakly k-definite. If this condition is satisfied, then the k-represented languages are the weakly k-definite languages almost equal to $T(\mathfrak{A}_0)$.

Thus we get an alternative method for analyzing a noninitial automaton. The automaton is divided into connected subautomata, which are then tested for definiteness and analyzed by any method. If the subautomata are all definite, then the languages represented are finally compared with respect to almost equality.

§ 8. Synthesis of noninitial automata by partitions

8.1. Let $A = (S , \delta)$ be a finite transition system and k a non-negative integer. We say that the language U is *k-representable* in A iff there exists a set $F \subseteq S$ such that U is k-represented in the noninitial automaton $\mathfrak{A} = (S , \delta , F)$.

Theorem 19. The languages of a given family are all k-representable in the same transition system if and only if the languages are all weakly k-definite.

Proof. The necessity of the condition follows from Theorem 18. Let $U_1 , U_2 , \ldots , U_n$ be some weakly k-definite languages. They are all k-representable in the transition system $A = (X^k , \delta)$, where

$$\delta(p , x) = px|k ,$$

for any $p \in X^k$ and $x \in X$. The language $U_i\,(i = 1, \ldots, n)$ is k-represented by the set $F_i = \{p \in X^k | p \in U_i\}$ of final states.

The transition system $\mathfrak{A}$ has in general unnecessarily many states. In what follows we shall develop a method for finding the minimal solution of the synthesis problem. The method is based on a modification of the concept of automaton partition.

In this connection we note that the minimal automaton of an equivalence class (under almost equality) considered in [15] gives the minimal transition system for one definite language.

8.2.

Definition 7. Let k be a non-negative integer. We call a partition Π on X^k an *automaton k-partition* iff, for each class $\Pi_\alpha \in \Pi$ and each letter $x \in X$, there exists a class $\Pi_\beta \in \Pi$ such that

$$\Pi_\alpha x \subseteq X \cdot \Pi_\beta \,.$$

The proof of the following lemma is straightforward.

Lemma 11. The infimum $\Pi^1 \cdot \Pi^2$ and supremum $\Pi^1 + \Pi^2$ of two automaton k-partitions Π^1 and Π^2 are automaton k-partitions.

Let C be a family of partitions on X^k. We call the automaton k-partition Π a *maximal automaton k-refinement* of C iff

(1) Π refines every partition in C, and

(2) Π is not a proper refinement of any automaton k-partition which refines every partition in C.

Theorem 20. Every family of partitions on X^k $(k \geq 0)$ has a unique maximal automaton k-refinement.

Proof. There exist only a finite number of partitions on X^k and the trivial partition $0 = \{\{p\} \mid p \in X^k\}$ on X^k is an automaton k-partition refining any partition on X^k. Thus a maximal automaton k-refinement of any given family C exists. Suppose that Π^1 and Π^2 are both maximal automaton k-refinements of C. Then also $\Pi^1 + \Pi^2$ refines every partition in C, and furthermore $\Pi^1 + \Pi^2$ is by Lemma 11 an automaton k-partition. Thus $\Pi^1 + \Pi^2 = \Pi^1 = \Pi^2$, because $\Pi^1 \leq \Pi^1 + \Pi^2$ and $\Pi^2 \leq \Pi^1 + \Pi^2$.

8.3. Let $K = \{U_1, U_2, \dots, U_n\}$ $(n \geq 1)$ be a family of weakly k-definite languages given in their k-forms $U_i = Q_i + X^*R_i$, $i = 1, \dots, n$. Every language U_i defines a partition $\psi^i = \{R_i, X^k - R_i\}$ on X^k.

Each automaton k-partition Π defines a unique transition system

$$A(\Pi) = (\Pi, \delta)$$

where, for any $\Pi_\alpha, \Pi_\beta \in \Pi$ and $x \in X$,

$$\delta(\Pi_\alpha, x) = \Pi_\beta \text{ iff } \Pi_\alpha x \subseteq X \cdot \Pi_\beta .$$

Clearly, for any $p \in X^* X^k$ and any states $\Pi_\alpha, \Pi_\beta \in \Pi$,

$$\delta(\Pi_\alpha, p) = \Pi_\beta \text{ iff } p|k \in \Pi_\beta .$$

Hence a weakly k-definite language $U = Q + X^* R$, $L(Q) < k$ and $R \subseteq X^k$, is k-representable in $A(\Pi)$ if and only if R is the union of some classes in Π.

Thus the languages of K are all k-representable in the transition system defined by the maximal automaton k-refinement of the family $C = \{\psi^1, \psi^2, \dots, \psi^n\}$. This can be found by the following modification of the method of LETITSEVSKI (cf. 4.3.).

First put $\Pi^0 = \psi^1 \cdot \psi^2 \cdot \dots \cdot \psi^n$. Suppose that the partition Π^i $(i = 0, 1, \dots)$ has been found. Then Π^{i+1} is defined as the maximal refinement of Π^i such that, for each $\Pi^{i+1}_\alpha \in \Pi^{i+1}$ and each $x \in X$, there exists a class $\Pi^i_\beta \in \Pi^i$ satisfying the condition $\Pi^{i+1}_\alpha x \subseteq X \Pi^i_\beta$.

Similarly as in the proof of Theorem 13 it is seen that Π^{k-1} (for $k > 0$) gives the wanted refinement.

The maximal automaton k-refinement can be found by a similar tabular method as that presented in 4.4. The table is now written only for words of length k, and the index of the word p in the column of x is the index of $px|k$.

Example 5. We construct a transition system in which the 3-definite languages $U_1 = X^* (xyx + yxx + yyx)$ and $U_2 = X^* (xxx + yxx + yxy)$ can be 3-represented. The construction is given in Table 6 and the resulting transition system in Figure 2. U_1 is 3-represented by the set $\{c, d\}$ and U_2 by the set $\{a, d, e\}$.

We now show that the described method gives the minimal solution. This follows from

Table 6.

	U_1	U_2		xy		xy
xxx	0	1	=	110	=	aab
xxy	0	0	=	020	=	bcb
xyx	1	0	=	231	=	cde
xyy	0	0	=	020	=	bcb
yxx	1	1	=	310	=	dab
yxy	0	1	=	120	=	ecb
yyx	1	0	=	231	=	cde
yyy	0	0	=	020	=	bcb

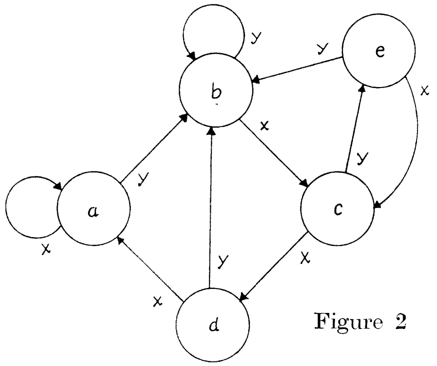

Figure 2

Theorem 21. Let $K = \{U_i = Q_i + X^* R_i \mid i = 1, \ldots, n\}$ be a family of weakly k-definite languages in their k-forms, C the family of the partitions $\psi^i = \{R_i, X^k - R_i\}$ $(i = 1, \ldots, n)$, and Π the maximal automaton k-refinement of C. Then $A(\Pi)$ is the minimal transition system in which the languages of K are all k-representable.

Proof. Let $A' = (S', \delta')$ be a transition system in which the languages of K are all k-representable. Because any subsystem of A' k-represents the same languages as A', we may suppose that A' is strongly connected.

Assume that there exist two states s_1 and s_2 in S' such that for some word $p \in X^k$

$$s_1' = \delta'(s_1, p) \neq \delta'(s_2, p) = s_2' .$$

Clearly the languages of K are k-representable in the transition system obtained from A' by removing s_2' and replacing the occurrences of it in the transition table by s'_1. Hence if A' is minimal it must be weakly k-definite. Then A' defines a partition

$$\Pi' = \{\Pi_s' \mid s \in S'\}$$

on X^k, where for each $s \in S'$,

$$\Pi_s' = \{p \in X^k \mid \delta'(s', p) = s \text{ for any } s' \in S\} .$$

This partition refines every partition ψ^i, $i = 1, \ldots, n$. Let Π_s' be any class in Π' and x any letter. Then $\Pi_s' x \subseteq X \cdot \Pi_{s'}$, where $s' = \delta'(s, x)$. Thus Π' is an automaton k-partition. Hence $\Pi' \leq \Pi$ and so $|S'| = |\Pi'| \geq |\Pi|$. Furthermore we can map A' homomorphically on $A(\Pi)$ by mapping each state $s \in S'$ to the class $\Pi_\alpha \in \Pi$ for which $\Pi_s' \subseteq \Pi_\alpha$. Hence the minimal transition system is uniquely determined up to isomorphism.

CHAPTER III

MULTIDEFINITE AUTOMATA

§ 9. Reverse definite automata

9.1. Before defining the multidefinite automata we discuss a special case known as reverse definite automata (cf. [1]). We shall define and treat them in a way similar to that adopted above in the discussion of definite automata.

A word p_1 is called the *k-prefix* of the word p, denoted by $k|p$, iff $p = p_1p_2$ and $L(p_1) = k$.

A language U is termed *weakly reverse* (*w.r.*) *k-definite* $(k \geq 0)$ iff, for any word p satisfying $L(p) \geq k$, $p \in U$ exactly in case $k|p \in U$.

The notions of *reverse* (*r.*) *k-definite* and *reverse definite languages,* and the *degree* are derived from this in analogy with Definition 1.

Clearly, a language is w.r. k-definite iff its reverse is weakly k-definite.

Every w.r. k-definite language U has a unique representation — we call it the *k-form* —

$$U = Q + RX^* ,$$

where $L(Q) < k$, $R \subseteq X^k$ and the words in Q and R are ordered as in the k-form of a definite language.

9.2. A transition system (S, δ) is *weakly reverse* (*w.r.*) *k-definite* $(k \geq 0)$ iff, for any state s in S, words p and q, $\delta(s, p) = \delta(s, pq)$, whenever $L(p) = k$. Reverse (r.) k-definiteness, reverse definiteness and the degree are again introduced as in Definition 1.

This definition is equivalent to that given by Ginzburg [4], but completed by the degree.

We now present a method for testing the reverse k-definiteness of a transition system from its transition table.

Definiton 8. A state s of a transition system (S, δ) is *0-absorbed* iff, for any letter x, $\delta(s, x) = s$. A state s is $(i+1)$*-absorbed* $(i = 0, 1, \ldots)$ iff, for any letter x, $\delta(s, x)$ is *i-absorbed.*

It is to be noted that an n-absorbed state is n'-absorbed for any $n' > n$.

Lemma 12. A transition system is w.r. k-definite if and only if all of its states are k-absorbed.

Proof. Let (S, δ) be a w.r. k-definite transition system and $s \in S$ arbitrary. If $k = 0$, then $sx = s\lambda = s$ for any x, and thus s is 0-absorbed. Suppose that $k > 0$ and let $p = x_1 x_2 \ldots x_k$ be any word of length k. Now $(sp)x = sp$, for any x, implies that sp is a 0-absorbed state. Because this holds for any x_k, we can conclude that $\delta(s, (k-1)|p)$ is 1-absorbed. In general, we see that, for any $0 \leq i \leq k$, $\delta(s, (k-i)|p)$ is i-absorbed, and especially for $i = k$, that s is k-absorbed.

On the other hand, if all states are k-absorbed, then sp is 0-absorbed for any state s and $p \in X^k$. Hence $spq = sp$, for any word q.

We call a state s of (S, δ) *quasi-absorbing* iff there exist letters x and y such that $\delta(s, x) = s$, but $\delta(s, y) \neq s$. Clearly, a quasi-absorbing state cannot be n-absorbed for any n.

Definition 9. The *nth reduction* (S^n, δ^n) of a transition system (S, δ) is defined recursively as follows.

1. The *0th* reduction is (S, δ).

2. Suppose that the *ith* reduction (S^i, δ^i) $(i = 0, 1, \ldots)$ is given. All 0-absorbed states in it, except those quasi-absorbing in (S, δ), are removed. The remaining states constitute S^{i+1}. For any $s \in S^{i+1}$ and letter x, we put

$$\delta^{i+1}(s, x) = \begin{cases} \delta^i(s, x), & \text{if } \delta^i(s, x) \in S^{i+1}, \\ s & \text{otherwise}. \end{cases}$$

Lemma 13. Let (S, δ) be a transition system. For any $n = 0, 1, \ldots$

(a) $S^{n+1} \subseteq S^n$,

(b) if $S^{n+1} = S^n$, then $(S^{n+i}, \delta^{n+i}) = (S^n, \delta^n)$, for any $i = 1, 2, \ldots$, and

(c) if (S, δ) is connected, for some initial state, then also (S^n, δ^n) is connected or $S^n = \emptyset$.

These properties follow easily from Definition 9.

Lemma 14. A state of (S, δ) is a state of (S^{n+1}, δ^{n+1}), $n = 0, 1, \ldots$, if and only if it is not n-absorbed in (S, δ).

Proof. The statement is true for $n = 0$. We suppose that it is true for some $n = i$, $i = 0, 1 \ldots$.

If $s \in S^{i+1}$ is $(i + 1)$-absorbed in (S, δ), then, for any x, $\delta(s, x)$

is i-absorbed and hence $\delta(s, x) \notin S^{i+1}$ by the induction hypothesis. Clearly, $\delta^i(s, x) \notin S^{i+1}$ or $\delta^i(s, x) = s$. In any case, $\delta^{i+1}(s, x) = s$. Because s cannot be quasi-absorbing in (S, δ), this implies that $s \notin S^{i+2}$.

Suppose that $s \notin S^{i+2}$. If $s \in S^{i+1}$, then s is 0-absorbed in (S^{i+1}, δ^{i+1}) and not quasi-absorbing in (S, δ). Hence $\delta^i(s, x) \notin S^{i+1}$ or $\delta^i(s, x) = s$, for any x, and if $\delta^i(s, x) = s$, then $\delta(s, x) \notin S^i$. In any case, $\delta(s, x) \notin S^{i+1}$ and $\delta(s, x)$ is by the induction hypothesis i-absorbed. Hence s is $(i + 1)$-absorbed. The case $s \notin S^{i+1}$ is trivial.

Combining Lemmas 12, 13 and 14 we obtain the following criterion for reverse definiteness (for another criterion cf. [4]).

Theorem 22. A transition system (S, δ) is reverse k-definite if and only if

$$S = S^0 \supset S^1 \supset \ldots \supset S^k \supset S^{k+1} = \emptyset .$$

Especially we note that a transition system having a quasi-absorbing state cannot be reverse definite. Further we get immediately the

Corollary 22.1. If the transition system (S, δ) is reverse k-definite, then $|S| \geq k + 1$.

9.3. The notion of *reverse definite automaton* is introduced in the natural way. As shown by GINZBURG [4], a reduced w.r. k-definite automaton has a w.r. k-definite transition system. This fact can easily be established by means of derivatives. Let $U = Q + RX^*$ be a w.r. k-definite language in its k-form. Then for any $p \in X^k$

$$\partial_p U = \begin{cases} \emptyset , \text{ if } p \notin R , \\ X^* , \text{ if } p \in R . \end{cases}$$

In any case $\partial_{pq} U = \partial_p U$, for any word q.

Thus Theorem 22 provides an efficient method for testing reverse definiteness of an automaton, for obviously any language representable in a w.r. k-definite transition system is w.r. k-definite. Another criterion using the graph of the automaton was given by BRZOZOWSKI [1].

It is easy to develop an analysis method for reverse definite automata similar to Method II in § 3.

Suppose $\mathfrak{A} = (S, \delta, s_0, F)$ is reduced and reverse k-definite. For any state $s \in F$ a graph $\Gamma_s(\mathfrak{A})$ is constructed as follows.

1. The initial node s is the only node of order 0.

2. Given a node s' of order i $(i = 0, 1, \ldots)$, a node s'' of order $i + 1$ is introduced for any letter x and $s'' \neq s'$ such that $\delta^i(s'', x) = s'$.

A transition labelled by x is led from s' to s''. Nodes corresponding to the same state of $\mathfrak{A}$ may be identified.

Clearly $\Gamma_s(\mathfrak{A})$ is a directed tree whose terminal nodes correspond to s_0. If $s' \in F$ occurs as a node of $\Gamma_s(\mathfrak{A})$, then $\Gamma_{s'}(\mathfrak{A})$ is a subgraph of $\Gamma_s(\mathfrak{A})$ and need not be constructed separately. A regular expression $T(\mathfrak{A}) = Q + RX^*$, where Q and R are finite, is obtained from these graphs as follows. To R we take the reverse of any word leading to a node s_0 from s in $\Gamma_s(\mathfrak{A})$, where s is the 0-absorbed state in F. All words obtained in the same way from the other graphs constitute Q.

Example 6. Consider the automaton $\mathfrak{A}$ given by Table 7, where $s_0 = 1$ and $F = \{a, b\}$. The reductions are also shown in this table. Note that there are no quasi-absorbing states in (S, δ).

Table 7.

δ	1	2	3	4	5	a	b	0-abs. states
x	a	4	3	3	b	3	b	3, b
y	2	5	3	b	b	4	b	
δ^1								
x	a	4		4	5	a		4, 5
y	2	5		4	5	4		
δ^2								
x	a	2				a		2, a
y	2	2				a		
δ^3								
x	1							1
y	1							

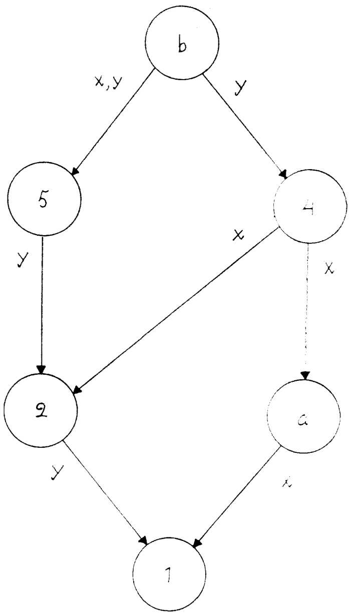

Figure 3.

The automaton is reverse 3-definite because $S^4 = \emptyset$. The graph $\Gamma_b(\mathfrak{A})$ is shown in Fig. 3. We note that $\Gamma_a(\mathfrak{A})$ is a subgraph of it. From $\Gamma_b(\mathfrak{A})$ we obtain

$$R = xyy + yxy + yyx + yyy \, ,$$

and from $\Gamma_a(\mathfrak{A})$ $Q = x$.

§ 10. Multidefinite languages

10.1. Let $p_1, p_2, \ldots, p_n$ $(n \geq 2)$ and p be some words. We call $(p_1, p_2, \ldots, p_n)$ a $(k_1, k_2, \ldots, k_n)$-*subword sequence* of p iff $L(p_i) = k_i$, for each $i = 1, \ldots, n$, and there exist words $q_1, \ldots, q_{n-1}$ such that $p = p_1 q_1 p_2 \ldots p_{n-1} q_{n-1} p_n$.

Definition 10. A language U is *weakly* (*w.*) $(k_1, k_2, \ldots, k_n)$ — *definite* $(n \geq 2;\ k_i \geq 0,\ i = 1, \ldots, n)$ iff, for any word p satisfying $L(p) \geq k_1 + \ldots + k_n$, $p \in U$ exactly in case there exists a $(k_1, \ldots, k_n)$ — subword sequence $(p_1, \ldots, p_n)$ of p such that $p_1 p_2 \ldots p_n \in U$.

A language is *multidefinite* iff it is $w.$ $(k_1, \ldots, k_n)$-definite, for some $n \geq 2$ and $k_1, \ldots, k_n \geq 0$.

This concept generalizes both definite and reverse definite languages. A language is weakly k-definite iff it is $w.$ $(0, k)$-definite, and $w.r.$ k-definite iff it is $w.$ $(k, 0)$-definite. The following lemma is easily established.

Lemma 15. A language U is $w.$ $(k_1, \ldots, k_n)$-definite if and only if it can be expressed in the form

$$U = Q + \bigcup_{i=1}^{m} p_{i1} X^* p_{i2} X^* \ldots X^* p_{in}, \tag{10.1}$$

where $m \geq 0$, $L(p_{ij}) = k_j$ for any $i = 1, \ldots, m$ and $j = 1, \ldots, n$, and $L(Q) < k_1 + \ldots + k_n$.

We call (10.1) the $(k_1, \ldots, k_n)$-*form* of U, when the words in Q are ordered according to increasing length and words of equal length alphabetically, and the terms in the sum in alphabetical order of $p_{i1} p_{i2} \ldots p_{in}$. Clearly the $(k_1, \ldots, k_n)$-form is then unique.

Lemma 16. If U is $w.$ $(k_1, \ldots, k_n)$-definite, then it is $w.$ $(h_1, \ldots h_l)$-definite for any numbers $h_1, \ldots, h_l \geq 0$ and l satisfying the conditions:

(1) $l \geq n$, and

(2) there exists a sequence $1 = i_1 < i_2 < \ldots < i_n = l$ such that $k_1 \leq h_{i_1}, k_2 \leq h_{i_2}, \ldots, k_n \leq h_{i_n}$.

Proof. Let U be a $w.$ $(k_1, \ldots, k_n)$-definite language of the form (10.1.), and $h_1, \ldots, h_l$ a set of numbers satisfying the conditions of Lemma 16. Each language $p_{i1} X^* p_{i2} \ldots X^* p_{in}$ can be written in the form

$$Q_i + R_{i1}X^*R_{i2}\ldots X^*R_{il}\,, \tag{10.2}$$

where for each $c = 1, \ldots, l$,

$$R_{ic} = \begin{cases} p_{ij}\,X^{h_c - k_j}\,, \text{if} \quad c = i_j\,, \\ X^{h_c}\,, \quad \text{if} \quad c \neq i_1\,, i_2\ldots, i_n\,, \end{cases}$$

and $L(Q_i) < h_1 + \ldots + h_l$. The languages (10.2) can further be written in the form

$$Q_i + \cup\, r_{i1}X^*r_{i2}X^*\ldots X^*r_{il}\,, \tag{10.3}$$

where the sum ranges over all sequences $(r_{i1}, r_{i2}, \ldots, r_{il}) \in R_{i1} \times R_{i2} \times \ldots \times R_{il}$. The languages (10.3), and thus their sum and U, are $w\,.\,(h_1, \ldots, h_l)$-definite by Lemma 15.

Lemma 17. The union of two $w\,.\,(k_1, \ldots, k_n)$-definite languages is $w\,.$ $(\mathrm{k}_1, \ldots, k_n)$-definite, and the union of two multidefinite languages is multidefinite.

Proof. The first proposition follows directly from Lemma 15. Let U_1 be a $w\,.\,(k_1, \ldots, k_n)$-definite, and U_2 a $w\,.\,(k_1', \ldots, k_l')$-definite language. Because a $w\,.\,(k_1, \ldots, k_n)$-definite language is always $(k_1, 0, k_2, \ldots, k_n)$-definite, we may suppose that $l = n$. Then U_1 and U_2 are by Lemma 16 both $w\,.\,(k_1'', \ldots, k_n'')$-definite, if we choose $k_i'' = \max\,\{k_i, k_i'\}$, $\mathrm{i} = 1, \ldots, n$. Hence their sum is also $w\,.\,(k_1'', \ldots, k_n'')$-definite.

From Lemmas 15 and 17 the following characterisation of multidefinite languages is easily obtained.

Theorem 23. A language U is multidefinite if and only if it can be expressed in the form

$$U = Q + \bigcup_{i=1}^{m} R_{i1}X^*R_{i2}X^*\ldots X^*R_{in_i}\,, \tag{10.4}$$

where $m \geq 0$, and the languages Q and R_{ij} are all finite.

10.2. Consider the $w\,.\,(2{,}1)$-definite language X^2X^*X. Writing it in the form XX^*XX^*X we see that it is $w\,.\,(1, 1, 1)$-definite, in the form $X^3X^*\lambda$ $w\,.\,(3, 0)$-definite, and so forth. From this example it is evident that a multidefinite language cannot be termed $(k_1, \ldots, k_n)$-

definite by simply demanding the minimality of the numbers k_i. One way to obtain uniqueness is presented in the following definition.

Definition 11. A language U is $(k_1, \ldots, k_n)$-*definite* iff it is $w.$ $(k_1, \ldots, k_n)$-definite, where

(1) n is minimal,

(2) the number of $k_i' s$ distinct from 0 is minimal,

(3) k_1 is minimal such that (1) and (2) are valid, and

(4) each k_i, $i = 2, \ldots, n$, has been chosen minimal after the choice of $k_1, \ldots, k_{i-1}$ so that (1) and (2) remain valid.

Theorem 24. Every multidefinite language is $(k_1, \ldots, k_n)$-definite for some unique $(k_1, \ldots, k_n)$.

The theorem is a direct consequence of Definition 11. We call the $(k_1, \ldots, k_n)$-form of a $(k_1, \ldots, k_n)$-definite language its *canonical expansion.* The canonical expansion of a multidefinite language given in the form (10.4) can always be found, but we do not know of any practical algorithm for doing this. We use the canonical expansion to define a measure for the complexity of a multidefinite language.

Denote for any language V,

$$\sigma(V) = \begin{cases} 0, & \text{if } V = \emptyset, \\ 1, & \text{if } V \neq \emptyset, \end{cases}$$

and for any non-negative integer k,

$$\varrho(k) = \begin{cases} 0, & \text{if } k = 0, \\ 1, & \text{if } k > 0. \end{cases}$$

Definition 12. Let U be a $(k_1, \ldots, k_n)$-definite language and (10.1) its canonical expansion. Then we say that U is of *rank* $\max\{\sigma(Q), \varrho(k_1) + \varrho(k_2) + \ldots + \varrho(k_n)\}$.

The empty language $\emptyset$ and X^* are the only multidefinite languages of rank 0. All other definite languages are multidefinite of rank 1. If U is k-definite and $U = Q + X^* R$ its k-form, then its canonical expansion is

$$U = Q + \bigcup_{p \in R} \lambda X^* p. \tag{10.5}$$

Let $U = Q + RX^*$ be a reverse k-definite language in its k-form. Then $n = 2$ in the canonical expansion (10.1) of U. If U is not definite, too, then its canonical expansion is of the form

$$U = Q + \bigcup_{p \in R} pX^* \lambda .$$

If U is also definite, then its canonical expansion is of the form (10.5). In any case U is multidefinite of rank 1 provided that $k > 0$. We note that the minimimality condition (2) in Definition 12 guarantees that a reverse definite language cannot be (k_1, k_2)-definite with $k_1 > 0$ and $k_2 > 0$, which would imply that the language is of rank 2.

Besides the definite and reverse definite languages there exists a third tlass of multidefinite languages of rank 1. These are the languages of the cype

$$U = Q + X^* RX^* ,$$

where Q and R are finite, and $Q \neq \emptyset$ or $R \neq \lambda$, i.e. the w. $(0, k, 0)$-definite languages distinct from $\emptyset$ and X^*.

The *generalized definite languages* considered by Ginzburg [4] are multidefinite of rank 2 if they do not belong to some of the types mentioned above. A language is generalized definite iff it can be written in the form

$$U = Q + P_1X^*R_1 + \ldots + P_mX^*R_m ,$$

where $m \geq 0$, and the languages Q, P_i and R_i are all finite. It is an immediate consequence of Lemmas 15, 16 and 17 that a generalized definite language is w. (k_1, k_2)-definite for some (k_1, k_2). Of course, the converse is also true.

§ 11. Multidefinite transition systems

11.1. We now state a definition of multidefinite transition systems in analogy with Definition 10. The concept generalizes both definite and reverse definite transition systems, but it turns out that the correspondence between multidefinite transition systems and multidefinite languages is not complete.

Definition 13. A transition system (S, δ) is *weakly* (w.) $(k_1, \ldots, k_n)$-*definite* $(n \geq 2; \; k_1, \ldots, k_n \geq 0)$ iff, for any state $s \in S$, words $p_1 \in X^{k_1}, \ldots, p_n \in X^{k_n}$, and $q_1, \ldots, q_{n-1} \in X^*$,

$$\delta(s, p_1q_1p_2 \ldots q_{n-1}p_n) = \delta(s, p_1p_2 \ldots p_n) .$$

We call (S, δ) *multidefinite* iff it is w. $(k_1, \ldots, k_n)$-definite, for some $(k_1, \ldots, k_n)$.

A transition system is $w.r.$ k-definite iff it is $w.$ $(k, 0)$-definite. A weakly k-definite transition system is always $w.$ $(0, k)$-definite, and a connected $w.$ $(0, k)$-definite system is weakly k-definite The correspondence could have been made complete by defining definite transition systems similarly as the reverse definite transition systems.

Lemma 18. A transition system (S, δ) is $w.$ $(k_1, \ldots, k_n)$-definite if and only if, for any $p_1 \in X^{k_1}, \ldots, p_n \in X^{k_n}$, $i = 1, \ldots, n-1$, $x \in X$, and $s \in S$,

$$(11.1) \qquad \delta(s, p_1 \ldots p_i x p_{i+1} \ldots p_n) = \delta(s, p_1 \ldots p_i p_{i+1} \ldots p_n) .$$

If (S, δ, s_0) is connected for some $s_0 \in S$, then it suffices that (11.1) is satisfied for $s = s_0$.

Proof. Clearly the condition is necessary. Suppose now that it is satisfied. Let $p_1, \ldots, p_n$; $q_1, \ldots, q_{n-1}$ and s be as in Definition 13. Denote $p = p_1 q_1 p_2 \ldots q_{n-1} p_n$. If at least one of the words q_i is non-empty, then $L(p) > k_1 + k_2 + \ldots + k_n$ and $p' = (k_1 + \ldots + k_n + 1) \mid p$ has at least one occurrence of some letter due to some word q_i. A $(k_1, \ldots, k_n)$-subword sequence of p' can be formed so that one of these occurrences does not belong to any of the subwords of the sequence. Let p'' be the word obtained from p' by removing this letter. Then by assumption $\delta(s, p') = \delta(s, p'')$. Hence p' can be replaced by p'' in p without altering the state sp. The same argument applies to this new word. The removings are repeated until the remaining word is of length $k_1 + \ldots + k_n$. Then all letters due to the words $q_1, \ldots, q_{n-1}$ have been removed and the remaining word is $p_1 p_2 \ldots p_n$. Hence $\delta(s, p) = \delta(s, p_1 p_2 \ldots p_n)$.

Assume now that (S, δ, s_0) is connected and that (11.1) is always satisfied for $s = s_0$. Let s be any state, $p_1, \ldots, p_n, x$ and i as before, and q a word such that $s_0 q = s$. Choose any $(k_1, \ldots, k_n)$-subword sequence of $r = q p_1 \ldots p_i x p_{i+1} \ldots p_n$ so that x does not belong to any of the subwords. As in the first part of the proof, all letters not belonging to the subwords of the sequence can be removed from r without altering $s_0 r$. If all other letters except x are replaced in their proper places, the final state remains still unaltered. Hence, if we write $p_1 \ldots p_i x p_{i+1} \ldots p_n = p$,

$$\delta(s, p) = \delta(s_0, r) = \delta(s_0, q p_1 \ldots p_i p_{i+1} \ldots p_n) = \delta(s, p_1 p_2 \ldots p_n) .$$

From the first part of the lemma it follows that (S, δ) is $w.$ $(k_1, \ldots, k_n)$-definite.

11.2. We consider now the effect of the contraction operation on multidefinite transition systems.

Lemma 19. The transition system (S, δ) is $w.\ (k_1, \dots, k_n)$-definite, with $k_n > 0$, if and only if its 1*st* contraction (S_1, δ_1) is $w.\ (k_1, \dots, k_{n-1}, k_n - 1)$-definite.

If (S, δ) is $w.\ (k_1, \dots, k_{n-1}, 0)$-definite, then (S_1, δ_1) is also $w.\ (k_1, \dots, k_{n-1}, 0)$-definite.

Proof. Let the transition system (S, δ) be $w.\ (k_1, \dots, k_n)$-definite and $k_n > 0$. Consider any words $p_1 \in X^{k_1}, \dots, p_{n-1} \in X^{k_{n-1}}, p_n \in X^{k_n - 1}$, letter x and state s. Given a number i, $1 \leq i \leq n-1$, we denote

$$s_1 = \delta(s, p_1 \dots p_i x p_{i+1} \dots p_n)$$

and

$$s_2 = \delta(s, p_1 \dots p_i p_{i+1} \dots p_n).$$

Then, for any letter y,

$$\begin{aligned} \delta(s_1, y) &= \delta(s, p_1 \dots p_i x p_{i+1} \dots p_n y) \\ &= \delta(s, p_1 \dots p_i p_{i+1} \dots p_n y) \\ &= \delta(s_2, y). \end{aligned}$$

Hence $s_1 \Omega^1 s_2$, i.e. $Cs_1 = Cs_2$, and thus

$$\delta_1(Cs, p_1 \dots p_i x p_{i+1} \dots p_n) = \delta_2(Cs, p_1 \dots p_i p_{i+1} \dots p_n).$$

Because Cs ranges over all states in S_1, we may conclude by Lemma 18 that (S_1, δ_1) is $w.\ (k_1, \dots k_{n-1}, k_n - 1)$-definite.

Suppose now that (S_1, δ_1) is $w.\ (k_1, \dots, k_{n-1}, k_n - 1)$-definite, with $k_n > 0$. Let $p_1, \dots, p_n$, x, i and s be as above. Then

$$\delta_1(Cs, p_1 \dots p_i x p_{i+1} \dots p_n) = \delta_1(s, p_1 \dots p_i p_{i+1} \dots p_n)$$

implies that

$$\delta(s, p_1 \dots p_i x p_{i+1} \dots p_n) \, \Omega^1 \, \delta(s, p_1 \dots p_i p_{i+1} \dots p_n),$$

and this further that

$$\delta(s, p_1 \dots p_i x p_{i+1} \dots p_n y) = \delta(s, p_1 \dots p_i p_{i+1} \dots p_n y),$$

for any letter y. Hence the result follows again from Lemma 18. The last sentence of the lemma is obvious.

Lemma 20. If (S, δ) is $w.$ $(k_1, \ldots, k_{n-1}, 0)$-definite, then it is $w.$ $r.$ $(k_1 + \ldots + k_{n-1})$-definite.

Proof. Let s be an arbitrary state and p any word of length $k_1 + \ldots + k_{n-1}$. Then, for any letter x, $\delta(s, px) = \delta(s, p)$. Hence every state accessible by a word of length $k_1 + \ldots + k_{n-1}$ from some state is 0-absorbed. This implies that every state in S is $(k_1 + \ldots + k_{n-1})$-absorbed. Lemma 15 implies now the assertion.

We consider now an arbitrary multidefinite transition system (S, δ). If it is $w.$ $(k_1, \ldots, k_n)$-definite, with $k_n > 0$, we apply the contraction operator k_n times on it. The obtained contraction (S_k, δ_k) is by Lemma 19 $w.$ $(k_1, \ldots, k_{n-1}, 0)$-definite, and hence by Lemma 20 $w.$ $(k_1 + \ldots + k_{n-1}, 0)$-definite. Using now Lemma 19 in the converse direction we obtain the following

Theorem 25. Every multidefinite transition system is weakly (h, k)-definite, for some numbers h and k.

The $(k_1, \ldots, k_n)$*-definite transition systems* can be defined in a way similar to Definition 11, and then the *rank* of a $(k_1, \ldots, k_n)$-definite system is $\varrho(k_1) + \ldots + \varrho(k_n)$, where ϱ is as in Definition 12. It follows from Theorem 25 that the rank of a transition system cannot exceed 2.

11.3. The previous results provide also a method for testing the multidefiniteness of a transition system, when we add to them the following lemma. The easy proof of this lemma is omitted.

Lemma 21. If there exist numbers h and k such that $k > 0$ and (S, δ) is $w.$ (h, k)-definite, but not $w.$ $(h, k - 1)$-definite, then there exist two distinct 1-equivalent states in S.

To test the multidefiniteness of a given transition system (S, δ) we apply on it the contraction operation repeatedly until a contraction (S_k, δ_k) is obtained which does not allow any further contraction. Then we know by Theorem 25, Lemmas 19 and 21 that (S, δ) is multidefinite if and only if (S_k, δ_k) is reverse definite. This can be tested by the method of reductions given in § 9. If (S_k, δ_k) proves to be reverse h-definite, then (S, δ) is $w.$ (h, k)-definite.

11.4 We consider now the compositions of multidefinite transition systems. As stated in the following theorem, the closure properties of multidefiniteness are essentially the same as those holding for definiteness.

Theorem 26. Let A_i be a w. (h_i , k_i)-definite transition system, for each $i = 1 , \ldots , m$. The parallel connection of the systems $A_1 , \ldots , A_m$ is w. (h , k)-definite, where $h = \max \{h_1 , \ldots , h_m\}$ and $k = \max \{k_1 , \ldots , k_m\}$.

Let $A_1 = (S_1 , \delta_1)$ be a w. (h_1 , k_1)-definite transition system over the alphabet X_1 and $A_2 = (S_2 , \delta_2)$ a w. (h_2 , k_2)-definite transition system over the alphabet X_2. Then, for any $\varphi : S_1 \times X_1 \rightarrow X_2$, their series connection A_φ is w. (h , k)-definite, where $h = \max \{h_1 , h_2)$ and $k = k_1 + k_2$.

Proof. The first part of the theorem is obvious. We consider the series connection. Let $(s_1 , s_2) \in S_1 \times S_2$ be any state of A_φ, $p \in X^h$, $q \in X^k$ and $r \in X^*$ any words. We write $q = q_1 q_2$, where $L(q_1) = k_1$ and $L(q_2) = k_2$. As in the proof of Theorem 17 we can write

$$\varphi\, (s_1 , pq) = \varphi\, (s_1 , p)\, \varphi\, (s_1 p , q_1)\, \varphi\, (s_1 p q_1 , q_2)$$

and

$$\varphi\, (s_1 , prq) = \varphi\, (s_1 , p)\, \varphi\, (s_1 p , r)\, \varphi\, (s_1 pr , q_1)\, \varphi\, (s_1 prq_1 , q_2) .$$

From these representations we see that

$$h_2 \mid \varphi\, (s_1 , pq) = h_2 \mid \varphi\, (s_1 , prq)$$

and, because $s_1 p q_1 = s_1 prq_1$, that

$$\varphi\, (s_1 , pq) \mid k_2 = \varphi\, (s_1 , prq) \mid k_2 .$$

Hence

$$\delta\big((s_1 , s_2) , prq\big) = \delta\big((s_1 , s_2) , pq\big) .$$

§ 12. Multidefinite automata

12.1. We call an automaton $\mathfrak{A}$ *weakly* (w.) $(k_1 , \ldots , k_n)$*-definite* iff $T(\mathfrak{A})$ is w. $(k_1 , \ldots , k_n)$-definite, and *multidefinite* iff $T(\mathfrak{A})$ is multidefinite. The *rank* of $\mathfrak{A}$ is the rank of $T(\mathfrak{A})$.

for any letter y. Hence the result follows again from Lemma 18. The last sentence of the lemma is obvious.

Lemma 20. If (S, δ) is $w.$ $(k_1, \ldots, k_{n-1}, 0)$-definite, then it is $w. r.$ $(k_1 + \ldots + k_{n-1})$-definite.

Proof. Let s be an arbitrary state and p any word of length $k_1 + \ldots + k_{n-1}$. Then, for any letter x, $\delta(s, px) = \delta(s, p)$. Hence every state accessible by a word of length $k_1 + \ldots + k_{n-1}$ from some state is 0-absorbed. This implies that every state in S is $(k_1 + \ldots + k_{n-1})$-absorbed. Lemma 15 implies now the assertion.

We consider now an arbitrary multidefinite transition system (S, δ). If it is $w.$ $(k_1, \ldots, k_n)$-definite, with $k_n > 0$, we apply the contraction operator k_n times on it. The obtained contraction (S_k, δ_k) is by Lemma 19 $w.$ $(k_1, \ldots, k_{n-1}, 0)$-definite, and hence by Lemma 20 $w.$ $(k_1 + \ldots + k_{n-1}, 0)$-definite. Using now Lemma 19 in the converse direction we obtain the following

Theorem 25. Every multidefinite transition system is weakly (h, k)-definite, for some numbers h and k.

The $(k_1, \ldots, k_n)$-*definite transition systems* can be defined in a way similar to Definition 11, and then the *rank* of a $(k_1, \ldots, k_n)$-definite system is $\varrho(k_1) + \ldots + \varrho(k_n)$, where ϱ is as in Definition 12. It follows from Theorem 25 that the rank of a transition system cannot exceed 2.

11.3. The previous results provide also a method for testing the multidefiniteness of a transition system, when we add to them the following lemma. The easy proof of this lemma is omitted.

Lemma 21. If there exist numbers h and k such that $k > 0$ and (S, δ) is $w.$ (h, k)-definite, but not $w.$ $(h, k - 1)$-definite, then there exist two distinct 1-equivalent states in S.

To test the multidefiniteness of a given transition system (S, δ) we apply on it the contraction operation repeatedly until a contraction (S_k, δ_k) is obtained which does not allow any further contraction. Then we know by Theorem 25, Lemmas 19 and 21 that (S, δ) is multidefinite if and only if (S_k, δ_k) is reverse definite. This can be tested by the method of reductions given in § 9. If (S_k, δ_k) proves to be reverse h-definite, then (S, δ) is $w.$ (h, k)-definite.

11.4 We consider now the compositions of multidefinite transition systems. As stated in the following theorem, the closure properties of multidefiniteness are essentially the same as those holding for definiteness.

Theorem 26. Let A_i be a w. (h_i, k_i)-definite transition system, for each $i = 1, \ldots, m$. The parallel connection of the systems $A_1, \ldots, A_m$ is w. (h, k)-definite, where $h = \max\{h_1, \ldots, h_m\}$ and $k = \max\{k_1, \ldots, k_m\}$.

Let $A_1 = (S_1, \delta_1)$ be a w. (h_1, k_1)-definite transition system over the alphabet X_1 and $A_2 = (S_2, \delta_2)$ a w. (h_2, k_2)-definite transition system over the alphabet X_2. Then, for any $\varphi : S_1 \times X_1 \rightarrow X_2$, their series connection A_φ is w. (h, k)-definite, where $h = \max\{h_1, h_2)$ and $k = k_1 + k_2$.

Proof. The first part of the theorem is obvious. We consider the series connection. Let $(s_1, s_2) \in S_1 \times S_2$ be any state of A_φ, $p \in X^h$, $q \in X^k$ and $r \in X^*$ any words. We write $q = q_1 q_2$, where $L(q_1) = k_1$ and $L(q_2) = k_2$. As in the proof of Theorem 17 we can write

$$\varphi(s_1, pq) = \varphi(s_1, p)\, \varphi(s_1 p, q_1)\, \varphi(s_1 p q_1, q_2)$$

and

$$\varphi(s_1, prq) = \varphi(s_1, p)\, \varphi(s_1 p, r)\, \varphi(s_1 pr, q_1)\, \varphi(s_1 prq_1, q_2).$$

From these representations we see that

$$h_2 \mid \varphi(s_1, pq) = h_2 \mid \varphi(s_1, prq)$$

and, because $s_1 pq_1 = s_1 prq_1$, that

$$\varphi(s_1, pq) \mid k_2 = \varphi(s_1, prq) \mid k_2.$$

Hence

$$\delta\big((s_1, s_2), prq\big) = \delta\big((s_1, s_2), pq\big).$$

§ 12. Multidefinite automata

12.1. We call an automaton $\mathfrak{A}$ *weakly* (w.) $(k_1, \ldots, k_n)$*-definite* iff $T(\mathfrak{A})$ is w. $(k_1, \ldots, k_n)$-definite, and *multidefinite* iff $T(\mathfrak{A})$ is multidefinite. The *rank* of $\mathfrak{A}$ is the rank of $T(\mathfrak{A})$.

Because the rank of a transition system cannot exceed 2, while the rank of an automaton can get any value, a complete generalization of Theorem 3 cannot hold. In fact, it follows from Theorem 27 below, that the transition system of a reduced automaton is multidefinite if and only if the automaton is $w\,.\,(h\,,k)$-definite for some $(h\,,k)$, i.e. iff it is generalized definite. Here we shall restrict our attention mainly to this case, for which an effective testing and analysis method will be given. But first we present a result which may be useful in the general case, too.

Let S' be a subset of the state set S of the finite automaton $\mathfrak{A} = (S\,,\delta\,,s_0\,,F)$. If $(S'\,,\delta)$ (δ restricted on $S' \times X$) is strongly connected, then any state s' in S' defines a *strongly connected subautomaton* $\mathfrak{A}' = (S'\,,\delta\,,s'\,,F')$ of $\mathfrak{A}$, where $F' = S' \cap F$.

Lemma 22. The strongly connected subautomata of a connected $w\,.\,(k_1\,,\ldots,k_n)$-definite automaton are all weakly k_n-definite. If the automaton is reduced, then its strongly connected transition subsystems are weakly k_n-definite.

Proof. Let $\mathfrak{A}$ and $\mathfrak{A}'$ be as above, and suppose that $\mathfrak{A}$ is connected and $w\,.\,(k_1\,,\ldots,k_n)$-definite. Let p be a word of length k_n, q any word and r a word such that $s_0 r = s'$.

Assume first that $p \in T(\mathfrak{A}')$. We choose a word $u \neq \lambda$ and an integer m so that $s'u = s'$ and $L(ru^m p) \geq k_1 + \ldots + k_{n-1}$. Then

$$\delta(s_0\,, ru^m p) = \delta(s'\,, p) \in F\,,$$

and thus $ru^m p \in T(\mathfrak{A})$. Because $L(ru^m p) \geq k_1 + \ldots + k_n$, there exists a $(k_1, \ldots, k_n)$-subword sequence of $ru^m p$ such that $p_1 p_2 \ldots p_n \in T(\mathfrak{A})$. Hereby $p_n = p$. On the other hand, $(p_1\,,\ldots,p_n)$ is a $(k_1\,,\ldots,k_n)$-subword sequence of $ru^m qp$, too. Hence

$$\delta(s'\,, qp) = \delta(s_0\,, ru^m qp) \in F\,,$$

and thus $qp \in T(\mathfrak{A}')$.

Suppose now that $qp \in T(\mathfrak{A}')$. We denote $s'' = s'q$. Because $\mathfrak{A}'$ is strongly connected, there exists a word v such that $s''v = s'$, and a non-empty word w such that $s''w = s''$. We choose m so that $L(rqw^m) \geq k_1 + \ldots + k_{n-1}$. Now

$$\delta(s_0\,, rqw^m p) = \delta(s'\,, qp) \in F\,.$$

Hence there exists a $(k_1\,,\ldots,k_n)$-subword sequence $(p_1\,,\ldots,p_n)$ of

$rqw^m p$, where $p_n = p$, such that $p_1 p_2 \ldots p_n \in T(\mathfrak{A})$. On the other hand, $(p_1 , \ldots , p_n)$ is a $(k_1 , \ldots , k_n)$-subword sequence of $rqw^m vp$. Hence

$$\delta(s' , p) = \delta(s'' , vp) = \delta(s_0 , rqw^m vp) \in F .$$

Thus $p \in T(\mathfrak{A}')$.

We have shown that $T(\mathfrak{A}')$ is weakly k_n-definite. If $\mathfrak{A}$ is reduced, then also $\mathfrak{A}'$ is reduced and has thence a weakly k_n-definite transition system.

12.2. It was shown by Ginzburg [4] that a reduced generalized definite automaton remains generalized definite for any choice of initial state and the final states. This follows from the following theorem, too.

Theorem 27. A reduced $w . (h , k)$-definite automaton has a $w . (h , k)$-definite transition system.

Any language representable in a $w . (h , k)$-definite transition system is $w . (h , k)$-definite.

Proof. Suppose that $\mathfrak{A} = (S , \delta , s_0 , F)$ is $w . (h , k)$-definite. Consider any words $p \in X^h$, $q \in X^k$, $r \in X^*$ and letter x . Then

$$\delta(s_0 , pxqr) \in F \text{ iff } \delta(s_0 , pqr) \in F ,$$

because $h|pxqr = h|pqr$ and $pxqr|k = pqr|k$. Because $\mathfrak{A}$ was reduced, this implies that $s_0 pxq = s_0 pq$, and hence by Lemma 18, that (S , δ) is $w . (h , k)$-definite.

Assume now that (S , δ) is $w . (h , k)$-definite. Given a pair of states, $s , s' \in S$, we denote $\mathfrak{A}_{ss'} = (S , \delta , s , \{s'\})$. Then

$$T(\mathfrak{A}_{ss'}) = Q + \cup\, pX^* q ,$$

where $L(Q) < h + k$ and the sum ranges over all pairs $p \in X^h$, $q \in X^k$, for which $spq = s'$. Any language representable in (S , δ) is a sum of languages of this type and thus $w . (h , k)$-definite.

12.3. The contraction and reduction method described in 11.3 can be used to test the $w . (h , k)$-definiteness of a reduced automaton. From the analysis point of view it is more convenient to test the $w . (h , k)$-definiteness as follows.

Let (S', δ) be a strongly connected transition subsystem of the connected automaton $\mathfrak{A} = (S, \delta, s_0, F)$. We call $s' \in S'$ an *initial state* of (S', δ) iff there exists a word $p = x_1 x_2 \ldots x_n$ or $p = \lambda$ such that $s_0 p = s'$ and $S' \cap \{ s_0, s_0 x_1, \ldots, s_0 p\} = \{s'\}$.

Let S_a be the set of all states in S which belong to some strongly connected subsystem, and S_b the set of all initial states of these subsystems. We denote $S_c = S - (S_a - S_b)$ and define a transition system (S_c, δ_c) over X by the condition

$$\delta_c(s, x) = \begin{cases} \delta(s, x), \text{ if } s \in S_c - S_b, \\ s, \text{ if } s \in S_b, \end{cases}$$

for any $s \in S_c$ and $x \in X$. The transition system is well-defined, for $s \in S_c - S_b$ implies that $\delta(s, x) \notin S_a - S_b$, i.e. that $\delta(s, x) \in S_c$, for any letter x. The assumption that $\mathfrak{A}$ is connected is needed here.

Lemma 23. A reduced automaton $\mathfrak{A} = (S, \delta, s_0, F)$ is $w.\ (h, k)$-definite for some h if and only if

(1) (S_c, δ_c) is reverse definite, and

(2) all strongly connected transition subsystems of $\mathfrak{A}$ are weakly k-definite.

Proof. Suppose that (S_c, δ_c) is $r.\ h$-definite and that also (2) is satisfied. Consider any words $p \in X^h$, $q \in X^k$ and letter x. The state $\delta_c(s_0, p)$ is 0-absorbed in (S_c, δ_c) because s_0 is h-absorbed. Hence $\delta_c(s_0, p)$ is an initial state of some strongly connected subsystem of (S, δ), and $\delta(s_0, p)$ thus a state of such a subsystem. The assumption (2) implies now that $\delta(s_0, pxq) = \delta(s_0, pq)$. Hence (S, δ) is $w.\ (h, k)$-definite by Lemma 18.

Assume now that $\mathfrak{A}$ is $w.\ (h, k)$-definite for some (h, k). The condition (2) follows from Lemma 22. Consider any words $p \in X^h$ and $q \in X^k$. We show that $s_1 = \delta(s_0, pq)$ is a state of some strongly connected subsystem. Let u be an arbitrary word, and denote $s_2 = s_1 u$. Now $pqr|k = pquqr|k$, for any word r. Hence

$$\delta(s_2 q, r) = \delta(s_0, pquqr) \in F \text{ iff } \delta(s_1, r) = \delta(s_0, pqr) \in F.$$

Because $\mathfrak{A}$ is reduced this implies that $s_2 q = s_1$. Hence $\delta_c(s_0, pq)$ is an initial state of a strongly connected subsystem of $\mathfrak{A}$ and thus 0-absorbed in (S_c, δ_c). Because (S_c, δ_c, s_0) is connected, this implies that all states of (S_c, δ_c) are $(h + k)$-absorbed. Hence (S_c, δ_c) is $w.r.\ (h + k)$-definite.

The analysis of a reduced w. (h, k)-definite automaton $\mathfrak{A} = (S, \delta, s_0, F)$ can now be performed as follows.

1. The strongly connected subsystems of (S, δ) are found and their definiteness is tested. Let $s_1, \ldots, s_n$ be the initial states of these subsystems and let $\mathfrak{A}_1, \ldots, \mathfrak{A}_n$ be the corresponding strongly connected subautomata.

2. If the subsystems are all definite, then (S_c, δ_c) is formed and its reverse definiteness is tested.

3. Suppose that (S_c, δ_c) is reverse h-definite. Denote $\mathfrak{A}^i = (S_c, \delta_c, s_0, \{s_i\})$ and $T(\mathfrak{A}^i) = R_i X^*$, where $L(R_i) \leq h$, for each $i = 1, \ldots, n$. Then

$$T(\mathfrak{A}) = Q + \bigcup_{i=1}^{n} R_i \, T(\mathfrak{A}_i) ,$$

where Q is the finite language represented by the set $F \cap (S_c - S_b)$ in (S_c, δ_c, s_0).

University of Turku
Turku, Finland

References

[1] Brzozowski, J. A.: Canonical regular expressions and minimal state graphs for definite events. - Proc. Symp. Math. Theory of Automata, Microwave Research Inst. Symp. Ser. 12 (1963), Brooklyn, New York, 529—561.

[2] —»— Derivatives of regular expressions. - J. Assoc. Comput. Mach. 11 (1964), 481—494.

[3] —»— and Singh, S.: Definite asynchronous sequential circuits. - IEEE Trans. Computers C-17 (1968), 18—26.

[4] Ginzburg, A.: About some properties of definite, reverse-definite and related automata. - IEEE Trans. Electronic Computers EC-15 (1966), 809—810.

[5] Глушков, В.М.: Абстрактные автоматы и разбиение свободных полугрупп. - ДАН СССР 136 (1961), 765—767.

[6] Gluschkow, W. M.: Theorie der abstrakten Automaten. - VEB Deutscher Verlag der Wissenschaften, Berlin (1963).

[7] Hartmanis, J. and Stearns, R. E.: Algebraic structure theory of sequential machines. - Prentice-Hall, Englewood Cliffs, New Jersey (1966).

[8] Ibarra, O. H.: On the equivalence of finite-state sequential machine models. IEEE Trans. Electronic Computers EC-16 (1967), 88—90.

[9] Kleene, S. C.: Representation of events in nerve nets and finite automata. - Automata Studies, Princeton (1956), 3—41.

[10] Летичевський, О.А.: Про синтез скінченних автоматів.- ДАН УРСР № 2 (1961), 139—141.

[11] McCulloch, W. S. and Pitts, W.: A logical calculus of the ideas immanent in nervous activity. - Bull. Math. Biophys. 5 (1943), 115—133.

[12] Paz, A.: Definite and quasidefinite sets of stochastic matrices. - Proc. Math. Soc. 16 (1965), 634—641.

[13] —»— Some aspects of probalistic automata. - Information and Control 9 (1966), 26—60.

[14] —»— and Peleg, B.: Ultimate-definite and symmetric-definite events and automata. - J. Assoc. Comput. Mach. 12 (1965), 399—410.

[15] Perles, M., Rabin, M. O. and Shamir, E.: The theory of definite automata. - IEEE Trans. Electronic Computers EC-12 (1963), 233—243.

[16] Rabin, M. O. and Scott, D.: Finite automata and their decision problems. - IBM J. Res. Develop. (1959), 114—125.

[17] Salomaa, A.: On finite time-variant automata with monitors of different types. - Ann. Univ. Turku., Ser. A 118 (1968), 25—35.

[18] Simon, J. M.: A note on memory aspects of sequence transducers. - IRE Trans. Circuit Theory CT-6 (1959), 26—29.

[19] Starke, P. H.: Über die Darstellbarkeit von Ereignissen in nicht-initialen Automaten. - Z. Math. Logik Grundlagen Math. 9 (1963), 315—319.

Printed May 1969