7. D. J. Rose, "Triangulated graphs and the elimination process," J. Math. Anal. Appl., 32, 527-609 (1970).
8. S. E. Markosyan and I. A. Karapetyan, "On perfect graphs," Dokl. AN ArmSSR, 13, No. 5, 292-296 (1976).

A SUBCLASS OF DETERMINISTIC CONTEXT-FREE LANGUAGES WITH A DECIDABLE INCLUSION PROBLEM

P. V. Gorshkov and A. B. Stavrovskii

UDC 519.713.2

A new subclass of deterministic context-free languages with a decidable inclusion problem is described in terms of finite automata on the direct product of free semigroups. The decidability proof exploits the structural properties of the automata.

This paper is a continuation of studies of finite automata on finitely defined semigroups and, in particular, of automata on direct products of free semigroups and groups as systems that define formal languages. Equality of the class of context-free (CF) languages and the class of languages accepted by finite automata on the direct product of a free semigroup and a free group was proved in [1] using some special word acceptance technique. In this technique, the language is defined as a certain subset of the transversal of a regular language (or a subset of the set of canonical forms of the words of a regular language) which is accepted by a finite automaton in the traditional sense. In this paper, we describe a subclass of deterministic CF languages (DCFLs) with a decidable language inclusion problem in terms of automata on the direct product of a free semigroup and a free group with certain structural constraints on the transition function.

The study of the class of DCFLs is of considerable theoretical and applied value, both because of their relation to program schemas [2, 3] and because of their uses in the description of programming language syntax [4, 5]. Characterization of subclasses of DCFLs with decidable language equality and inclusion problems plays an essential role in the study of the properties and the structure of the class of DCFLs [6]. The class of languages studied in this paper is a subclass of the class of DCFLs accepted by real-time deterministic pushdown store automata (DPDAs) with a finite number of generalized final states [7]. An essential feature of the languages of this class is that they do not necessarily have the prefix property [4]. This is due to the fact that this class contains nonregular languages that are not accepted by the store-emptying superdeterministic DPDAs of [8]. The class of languages accepted by the store-emptying superdeterministic DPDAs contains a subclass of languages which so far is the largest known class of nonregular languages with a decidable inclusion problem. The class of languages studied in this paper is thus a new subclass of DCFLs with a decidable inclusion problem.

The analysis of the inclusion problem in this class of languages essentially relies on their definition in terms of finite automata on the direct product of a free semigroup and a free group. With this definition of the languages, we can investigate their properties using the structural properties and the finiteness of the automaton state set, as well as the structural properties of the words in the alphabet of the free group generators.

The results are presented in the following way. Section 1 defines the direct mixed semigroup which up to isomorphism coincides with the direct product of a free semigroup and a free group. As a result of this coincidence, direct mixed semigroups are identified with the corresponding direct products. For a finite automaton A, we define the traditional concepts of path, the set of paths of the automaton PP(A), the set of its initial final paths IPF(A), and the path concatenation operation on the set PP(A). We also define a path label as a word in some alphabet and different forms of path labels, which are obtained from path labels by various combinations of relationships. In terms of these forms of path labels, we define the language K_F-accepted by a finite automaton on a direct mixed subgroup.

A deterministic separated unary automaton (DSUA) is defined as an automaton on a direct mixed semigroup with certain constraints on the transition function.

Translated from Kibernetika, No. 2, pp. 12-17, March-April, 1990. Original article submitted November 16, 1988.

0011-4235/90/2602-0156$12.50

Section 2 establishes the place of the class of languages K_F-accepted by DSUAs in the DCFL hierarchy (Theorem 1).

Section 3 introduces the notion of a paired automaton, a path in a paired automaton, path coordinates, and left stability of a paired automaton. We describe the construction of a paired automaton from a given pair of arbitrary DSUAs, which has the property of left stability if and only if the language K_F-accepted by the first DSUA is included in the language K_F-accepted by the second DSUA (Lemma 1). All subsequent constructions are needed in order to prove decidability of the left-stability property in the class of paired automata.

Section 4 introduces the key concepts of this study - a (left) ε-path in a paired automaton, the nesting relation on the set of ε-paths of the automaton, the nesting degree of an ε-path. We prove finiteness of the length of an ε-path with a finite nesting degree (Lemma 2).

Section 5 defines the property of ε-periodicity of a paired automaton on some set of its paths. Two state sets are defined for a paired automata. We show that a paired automaton is stable if and only if it is ε-periodic and these two state sets coincide (Lemma 3). We also show that neither ε-periodicity nor equality of the two state sets on its own is sufficient for stability of the automaton.

This stability criterion is formalized in terms of ε-periodicity on a finite set of ε-paths with a bounded nesting degree and therefore a bounded length. We also show that in order to test the two state sets for equality, it is sufficient to examine a finite set of automaton paths of finite length (Lemmas 4-6). This leads to Theorem 2 on decidability of the stability criterion of a paired automaton and Theorem 3 on decidability of the inclusion problem in the class of languages K_F-accepted by DSUAs.

1. BASIC DEFINITIONS

We use the popular method of defining the semigroup $H = (\Sigma, \Phi)$ by its generator system Σ and the system of defining relationships Φ [9]. Let $S = (\Sigma, \emptyset)$ be a free semigroup of words in the alphabet $\Sigma = \{a_1,...,a_{n_1}\}$, $G = (\Gamma, \Phi_G)$ a semigroup of words in alphabet $\Gamma = \{b_1,...,b_{n_2},b_1^{-1},...,b_{n_2}^{-1}\}$with the system of defining relationships $\Phi_G = \{b_ib_i^{-1} \simeq b_i^{-1}b_i \simeq \varepsilon, i = 1, ..., n_2\}$, where ε is the empty word. Systems of defining relationships of the form Φ_G are called group systems. On the set of words Γ^*, the system Φ_G induces a Φ_G-equality relation, denoted by $\simeq$, which is a congruence of the set Γ^*. In the semigroup G, an irreducible word is a word that has no subwords of the form $\sigma\sigma^{-1}$, $\sigma \in \Gamma$. Here $(\sigma^{-1})^{-1} = \sigma$. Clearly, each $\simeq$-equivalence class of the set Γ^* contains a unique irreducible word, and therefore as the transversal E_G of the semigroup G we naturally consider the set NF(Γ) of irreducible words on Γ. The word γ from NF(Γ) is called a canonical form of the word β from Γ^* and is denoted by nf(β) if $\gamma \simeq \beta$. The set NF(Γ) of canonical forms of words from Γ^* with the operation * defined on this set as $\alpha * \beta = \mathrm{nf}(\alpha\beta)$ is a free group with the generator set Γ and the unity ε [10].

Let $\Sigma \cap \Gamma = \emptyset$. A direct mixed semigroup $H = (\Delta, \Phi)$, or DM(1, 1)-semigroup, is a semigroup with the generator system $\Delta = \Sigma \cup \Gamma$ and a system of defining relations $\Phi = \Phi_G \cup \Psi$, where $\Psi = \{\sigma_1\sigma_2 \simeq \sigma_2\sigma_1 \mid \sigma_1 \in \Sigma, \sigma_2 \simeq \Gamma\}$.

The system Φ induces the Φ-equality relation, which we also denote by $\simeq$, on the set of words from Δ^*, and as the canonical form cf(γ) of the word γ from Δ^* we naturally take the form $\alpha\beta$, where $\alpha \in \Sigma^*$, $\beta \in$ NF(Γ), and $\beta = \mathrm{nf}(\beta')$, where the word $\alpha\beta'$ is obtained from the word γ by application of relationships only from Ψ. The word $\alpha\beta'$ will be denoted by sf(γ). The set of canonical forms of the words from Δ^* or the transversal of the semigroup H will be denoted by CF(H). It is easy to show that CF(H) is a semigroup and it coincides up to isomorphism with the direct product $H' = S \otimes G$ of the free semigroup S and the free group G. Therefore the DM(1, 1)-semigroups of the form H will be identified with the corresponding direct products.

A finite automaton (nondeterministic, with ε-transitions), is the object $A = (Q, X, q_0, \delta, F)$ that consists of the state set Q, the input alphabet X, the initial state q_0, the transition relation $\delta \subseteq Q \times X_\varepsilon \times Q$, where $X_\varepsilon = X \cup \{\varepsilon\}$, and the set of final states F, $F \subseteq Q$. All these sets are nonempty and finite.

A nondegenerate path of length r in the automaton A is an arbitrary sequence of form $\lambda = \langle p_0,\sigma_1,p_1,...,\sigma_r,p_r\rangle$, where $p_0 \in Q$, $(p_{k-1},\sigma_k,p_k) \in \delta$, $k = 1,...,r$, $r > 0$. The length of the path λ is denoted by $|\lambda|$. In what follows, we only consider paths of nonzero length, unless otherwise specified. A path of length 0 is called degenerate. A subpath of the path λ is an arbitrary path of the form $\langle p_i,\sigma_{i+1},p_{i+1},...,\sigma_j,p_j\rangle$, $0 \leq i < j \leq r$. For i = 0, the subpath is called a prefix of the path λ. A path λ is called initial when $p_0 = q_0$ and final when $p_r \in F$. By PP(A), IP(A), IPF(A) we denote respectively the set of all paths of the automaton A, the set of initial paths, and the set of initial final paths.

For the paths $\lambda_1 = \langle p_0, \sigma_1, p_1, ..., \sigma_r, p_r\rangle$, $\lambda_2 = \langle v_0, \tau_1, v_1, ..., \tau_s, v_s\rangle$ from PP(A) of an arbitrary automaton A, their concatenation $\lambda_1 * \lambda_2$ is $\langle p_0, ..., \sigma_r, p_r, \tau_1, v_1, ..., \tau_s, v_s\rangle$ for $v_0 = p_r$; otherwise, the result of concatenation is undefined.

The label $sl(\lambda)$ of the path $\lambda = \langle p_0,\sigma_1,p_1,...,\sigma_r,p_r\rangle$ is the word $\sigma_1...\sigma_r$ from X^*.

A DM(1, 1)-automaton is an automaton on the DM(1, 1)-semigroup $H = (\Delta, \Phi)$, i.e., an automaton whose alphabet coincides with Δ.

Let $A = (Q, \Delta, q_0, \delta, F)$ be a DM(1, 1)-automaton. If K_F-accepts the language $K_F(A) = \{\alpha \in \Sigma^* \mid \exists\ \lambda \in IPF(A)\ (cf\ (sl(\lambda)) = \alpha)\}$.

Let A be a DM(1, 1)-automaton, $\lambda \in PP(A)$, and $sf(sl(\lambda)) = \alpha\beta$. In this case, the word α is denoted by $w(\lambda)$, β by $gl(\lambda)$, and $nf(\beta)$ by $c(\lambda)$.

The DM(1, 1)-automaton $A = (Q, \Delta, q_0, \delta, F)$ is called a deterministic separated unary automaton (DSUA) if it satisfies the following constraints:

1) $Q = Q_1 \cup Q_2,\ Q_1 \cap Q_2 = \varnothing;$
2) $\delta \subset Q_1 \times \Sigma \times Q \cup Q_2 \times \Gamma_\varepsilon \times Q_1;$
3) $(p, \sigma, q) \in \delta \Rightarrow \forall q' \in Q\ (q' \neq q \Rightarrow (p, \sigma, q') \notin \delta);$
4) $(p,\ \sigma,\ q) \in \delta\ \&\ p \in Q_2 \Rightarrow [\forall q' \in Q\ \forall\ \sigma \in \Gamma_\varepsilon\ ((q' \neq q \vee \sigma' \neq \sigma) \Rightarrow (p, \sigma', q') \notin\ \delta)]\ \&\ (q \in Q_1);$
5) $q_0 \in Q_1;$
6) $F = Q_1.$

For DSUAs we use the notation $A = (Q_1, Q_2, \Sigma, \Gamma, q_0, \delta)$. States from the set Q_1 are called Σ-states, states from the set Q_2 are called Γ-states. Let $\lambda = \langle p_0,\sigma_1,p_1,\dots,\sigma_r,p_r\rangle$ be a path in a DSUA. The maximal subsequence of the sequence $p_0,p_1,\dots,p_r$ consisting of Σ-states is called the Q_1-trace of this automaton.

2. THE CLASS OF DSUA LANGUAGES

The set of languages K_F-accepted by DSUAs will be denoted by $\mathfrak{K}_F$ (DSU). We will show that this set is a subset of the class of languages accepted by real-time DPDAs with finitely many generalized final states [7]. The relevant terminological definitions for PDAs can be found in [4].

DPDAs are defined as objects of the form $B = (Q, \Sigma, \Gamma_B, q_0, Z_0, \delta, F)$, where Q is the set of states, Σ is the input alphabet, Γ_B is the alphabet of pushdown store symbols, $q_0 \in Q$ is the initial state, $Z_0 \in \Gamma_B$ is the initial pushdown store symbol, $\delta \subseteq Q \times \Sigma \times \Gamma \to Q \times \Gamma^*$ is the transition function, $F \subseteq Q \times \Gamma^*$ is the set of generalized final states, and all these sets are finite. Transitions from δ are denoted as $(q, a, X) \to (p, \gamma)$. These automata are also called DRF automata. A DRFA B accepts the input word α if $(q_0, \alpha, Z_0) \overset{*}{\underset{B}{\vdash}} (q, \varepsilon, \gamma)$, where $(q, \gamma) \in F$. The set of these words forms the language L(B) accepted by the automaton B.

Given an arbitrary DSUA A, we will construct a DRFA B such that $K_F(A) = L(B)$. Let $A = (Q_1, Q_2, \Sigma, \Gamma, q_0, \delta)$ be an arbitrary DSUA. Construct the DRFA $B = (Q', \Sigma, \Gamma', q_0', Z_0, \delta', F)$, where $Q' = Q_1 \cup \{q_0'\}$, $\Gamma' = \Gamma \cup \{Z_0\}$, $F' = Q_1 \times \{Z_0\}$, and the transition function δ' is constructed as follows:

1) if for some $p, q \in Q_1$, $p' \in Q_2$, $a \in \Sigma$, and $X \in \Gamma_\varepsilon$ we have (p, a, p'), $(p', X, q) \in \delta$, then for all $Z \in \Gamma'$, $Z \neq X^{-1}$ we have $(p, a, Z) \to (q, ZX) \in \delta'$, and also $(p, a, X^{-1}) \to (q, \varepsilon) \in \delta$;

2) if for some $p, q \in Q_1$ and $a \in \Sigma$ we have $(p, a, q) \in \delta$, then for all $Z \in \Gamma'$ we have $(p, a, Z) \to (q, Z) \in \delta'$;

3) for $a \in \Sigma$, $Z \in \Gamma$, $p \in Q_1$, and $\gamma \in \Gamma^*$ we have $(q_0', a, Z) \to (p, \gamma) \in \delta'$ if and only if $(q_0, a, Z) \to (p, \gamma) \in \delta'$;

4) there are no other transitions in δ'.

Note that the state q_0' is introduced in order to ensure that the automaton B does not accept ε, because by definition DSUAs do not K_F-accept ε.

It is easy to see that there is a computation $(q_0, \alpha, Z_0) \overset{*}{\vdash} (q, \varepsilon, Z_0)$, $q \in Q_1$ in the automaton B for an arbitrary $\alpha \in \Sigma^*$ if and only if the DSUA A K_F-accepts α. A formal proof will be found in [11]. Thus, $L(B) = K_F(A)$ and we have the inclusion $\mathfrak{K}_F(DSU) \subseteq \mathfrak{L}$ (DRF), where $\mathfrak{L}$ (DRF) denotes the class of languages accepted by DRFAs. The inclusion is clearly proper.

The class $\mathfrak{L}$ (SD_0) of languages accepted by store-emptying superdeterministic DPDAs contains a subclass of languages with a decidable inclusion problem, which is known to be the largest class of such languages [8].

We say that a language has the prefix property if any word in this language is not a subword of any other word in the language [4]. Since languages from the class $\mathfrak{L}$ (SD_0) are accepted by store-emptying DPDAs, they have the prefix property [5]. The class $\mathfrak{K}_F(DSU)$ contains languages that do not have the prefix property.

Consider the language $L_0 = \{\alpha \mid \alpha$ is nonempty and contains the same number of symbols a_1 and $a_2\}$ on the alphabet $\Sigma = \{a_1, a_2\}$. Clearly, $L_0 = K_F(A_0)$, $A_0 = (\{q_0\}, \{q_1, q_2\}, \Sigma, \{b, b^{-1}\}, q_0, \delta)$, where $\delta = \{(q_0, a_1, q_1), (q_0, a_2, q_2), (q_1, b, q_0), (q_2, b^{-1}, q_0)\}$. The language L_0 obviously does not have the prefix property.

THEOREM 1. $\mathfrak{K}_F(DSU) \not\subseteq \mathfrak{L}(DRF)$, $\mathfrak{K}_F(DSU) \not\subseteq \mathfrak{L}$ (SD_0).

3. PAIRED AUTOMATA

Let us define paired automata. Let $G_i = (\Gamma_i, \Phi_i)$, $i = 1, 2$, be the semigroups of words in the alphabets $\Gamma_i = \{b_{i1}, ..., b_{in_i}, b_{i1}^{-1}, ..., b_{in_i}^{-1}\}$ with group systems of relationships Φ_i. A paired (finite nondeterministic) automaton is the object $A = (Q, \Gamma_1, \Gamma_2, q_0, \delta)$, where Q is the finite state set, Γ_1, Γ_2 are the alphabets, $\delta \subseteq Q \times \Gamma_{1\varepsilon} \times \Gamma_{2\varepsilon} \times Q$ is the set of transitions.

The path λ of length $|\lambda| = m$, $m > 0$, in a paired automaton is a sequence of the form $\langle p_0, (\sigma_1, \tau_1), p_1, ..., (\sigma_m, \tau_m), p_m \rangle$, where $p_0 \in Q$, $(p_{i-1}, \sigma_i, \tau_i, p_i) \in \delta$, $i = 1, ..., m$. The trace of the path λ is the sequence $\langle p_0, p_1, ..., p_m \rangle$. Similarly to Section 1, we define degenerate and initial paths. The set of all paths and the set of all initial paths are denoted by PP(A) and IP(A), respectively.

The label $psl(\lambda)$ of the path λ is the pair of words $\langle \sigma_1 \ldots \sigma_m, \tau_1 \ldots \tau_m \rangle$ from $\Gamma_1^* \times \Gamma_2^*$. The word $\sigma_1 ... \sigma_m$ is denoted by $lsl(\lambda)$, and the word $\tau_1 ... \tau_m$ by $rsl(\lambda)$. By $c_1(\lambda)$ and $c_2(\lambda)$ respectively we denote $nf(lsl(\lambda))$ and $nf(rsl(\lambda))$. These are the left and right coordinates of the path λ.

The paired automaton A is called left-stable if $\forall \lambda \in IP(A)\ (c_1(\lambda) = \varepsilon \Rightarrow c_2(\lambda) = \varepsilon)$.

For an arbitrary pair of DSUAs, we will construct a paired automaton which is left-stable if and only if the language K_F-accepted by the first DSUA is included in the language K_F-accepted by the second DSUA. We assume that the DSUAs are everywhere defined on the symbols of the alphabet Σ. The last assumption can be made without loss of generality, because for any DSUA there is a K_F-equivalent DSUA which is everywhere defined on the symbols from Σ [11].

Let us informally describe the paired automaton A constructed from a pair of DSUAs A_1 and A_2. We can fix the partial mapping $\varphi \subseteq PP(A_1) \times PP(A_2) \rightarrow PP(A)$, which is defined on pairs starting and ending in Σ-states of the paths (λ_1, λ_2) such that $w(\lambda_1) = w(\lambda_2)$ and satisfies the condition $\forall \lambda_1, \lambda_2^* (\varphi(\lambda_1, \lambda_2) = \lambda \Rightarrow gl(\lambda_1) = lsl(\lambda) \& gl(\lambda_2) = rsl(\lambda))$. The states of the automaton A are the pairs of end states for pairs of paths from the domain of definition of the mapping φ and the initial state of the automaton A is the pair of the initial states of the automata A_1 and A_2. If $\lambda = \varphi(\lambda_1, \lambda_2)$, then the trace of the path λ consists of pairs of states that form Q_1-traces of the paths λ_1 and λ_2. This mapping φ is not one-to-one, and any path λ in the paired automaton is thus in general the image of some set of pairs of paths in the automata A_1 and A_2, but $gl(\lambda_1) = lsl(\lambda)$, $gl(\lambda_2) = rsl(\lambda)$, $w(\lambda_1) = w(\lambda_2)$ for each such pair λ_1, λ_2.

Formally, let $A_i = (Q_{i1}, Q_{i2}, \Sigma, \Gamma_i, q_{0i}, \delta_i)$, $i = 1, 2$ be arbitrary DSUAs. First we use them to construct the intermediate automata $A_i' = (Q_i', \Sigma_i', q_{0i}', \delta_i')$, $i = 1, 2$ with state sets $Q_i' = Q_{i1}$, alphabets $\Sigma_i' = \Sigma \times \Gamma_{i\varepsilon}$, initial states $q_{01}' = q_{0i}$, and the following transition sets δ_i'. If for some $q, p \in Q_{i1}$, $q' \in Q_{i2}$, $a \in \Sigma$, $b \in \Gamma_{i\varepsilon}$ we have (q, a, q'), $(q', b, p) \in \delta_i$ in the automaton A_i, then $(q, (a, b), p) \in \delta_i'$; if for some $q, p \in Q_{i1}$, $a \in \Sigma$ we have $(q, a, p) \in \delta_i$ in the automaton A_i, then $(q, (a, \varepsilon), p) \in \delta_i'$, and there are no other transitions in δ_i'.

Then given the pair of intermediate automata A_1', A_2', we construct the paired automaton $A = (Q', \Gamma_1, \Gamma_2, q_0', \delta)$:

1) $q_0' = [q_{01}', q_{02}']$;

2) as long as the automaton A does not contain the states and transitions listed below, they are added to this automaton by the following rule: if $[q_1, q_2] \in Q'$ and $(q_i, (a, b_i), p_i) \in \delta_i'$ for some $a \in \Sigma$, $b_i \in \Sigma_{i\varepsilon}$, $p_i \in Q_i'$, $i = 1, 2$, then the state $[p_1, p_2]$ is added to Q' and the transition $([q_1, q_2], b_1, b_2, [p_1, p_2])$ is added to δ'.

Since initial DSUAs are finite, this construction of the automaton A halts and is unique.

LEMMA 1 [11]. For arbitrary DSUAs A_1 and A_2, $K_F(A_1) \subseteq K_F(A_2)$ if and only if the corresponding paired automaton is left-stable.

4. NESTING OF PATHS IN PAIRED AUTOMATA

The proof of decidability of the left-stability property in the set of paired automata relies on the fact that in order to establish if a paired automaton is left-stable, it suffices to consider a certain set of its paths of finite length. Here we introduce some concepts connected with paths in paired automata, and these concepts are used in Section 5 to define the relevant finite sets of paths.

Let $A = (Q, \Gamma_1, \Gamma_2, q_0, \delta)$ be a paired automaton. Denote by PP(A, p, q) the set of all its paths from state p to state q. The elements of PP(A, p, q) are called (p, q)-paths, and the states p and q are the ends of these paths. Let $\lambda = \langle p_0, (\sigma_1, \tau_1), p_1, ..., (\sigma_m, \tau_m), p_m \rangle$, where $p_0 = p$, $p_m = q$ is an arbitrary (p, q)-path of length $|\lambda| = m$. A subpath of the path λ is a sequence of the form $\mu = \langle p_i, (\sigma_{i+1}, \tau_{i+1}), p_{i+1}, ..., (\sigma_j, \tau_j), p_j \rangle$, where $0 \leq i < j \leq m$. For $i = 0$, the path μ is called a prefix of the path λ. On the set of paths PP(A) of the automaton A we define the path nesting relation $\leq_p$: $\mu \leq_p \lambda \Leftrightarrow \mu$ is a subpath of the path λ. Clearly, $\leq_p$ is a partial order relation. The path μ is called a proper subpath of the path λ if $\mu \leq_p \lambda$ and μ is not equal to λ. Note that if $\mu <_p \lambda$, then $|\mu| < |\lambda|$.

For an arbitrary given pair $(p, q) \in Q^2$, we denote by $PP(\lambda, p, q)$ the set of all (p, q)-subpaths of the path λ, by $T(\lambda)$ the set of all ordered pairs of states (u, v) for which there exists a (u, v)-subpath of the path λ, i.e., $T(\lambda) = \{(u, v) \mid PP(\lambda, u, v) \neq \emptyset\}$.

By definition, $PP(A) = \bigcup_{p,q \in Q} PP(A, p, q)$, and the nesting relation $\leq_p$ induces the nesting relation $\leq_{p,q}$ on the sets $PP(A, p, q)$: $\mu \leq_{p,q} \lambda \Leftrightarrow$ the (p, q)-path μ is a subpath of the (p, q)-path λ. All the relations $\leq_{p,q}$, $p, q \in Q$, are clearly partial orders, and so are their restrictions to the sets $PP(\lambda, p, q)$ for arbitrary paths $\lambda \in PP(A)$.

For an arbitrary (u, v)-path λ from PP(A), denote by $d(\lambda, p, q)$ the maximum length of the maximum chain without unity in the partially ordered set $PP(\lambda, p, q)$ with the order $\leq_{p,q}$. By $d(\lambda)$ denote $\max\{d(\lambda, p, q) \mid (p, q) \in T(\lambda)\}$. This number is called the nesting degree of the path λ.

The path λ from PP(A) is called a (left) ε-path if $c_1(\lambda) = \varepsilon$. Similarly to the concepts of a subpath and nesting of paths, we define on the set $PP_\varepsilon(A)$ of all ε-paths of the automaton A the concept of an ε-subpath of an ε-path, a proper ε-subpath, and the ε-nesting relation $\leq_\varepsilon$. Note that if $\mu <_\varepsilon \lambda$, then $\mu <_p \lambda$, but in general the converse is not true.

Let $PP_\varepsilon(\lambda, p, q)$ be the set of all ε-(p, q)-subpaths of the ε-path λ, $T_\varepsilon(\lambda)$ the set $\{(p, q) \in Q^2 \mid PP_\varepsilon(\lambda, p, q) \neq \emptyset\}$. We similarly define the sets $PP_\varepsilon(A, p, q)$ and the relations $\leq_{\varepsilon,p,q}$.

Denote by $IP_\varepsilon(A)$ the set of initial ε-paths, by $T_\varepsilon(A)$ the set $\{(p, q) \in Q^2 \mid PP_\varepsilon(A, p, q) \neq \emptyset\}$.

For an arbitrary path λ from $PP_\varepsilon(A)$ we denote by $d_\varepsilon(\lambda, p, q)$ the maximum length of the maximum chain without unity in the partially ordered set $PP_\varepsilon(\lambda, p, q)$ with the order $\leq_{\varepsilon,p,q}$, and by $d_\varepsilon(\lambda)$ we denote $\max\{d_\varepsilon(\lambda, p, q) \mid (p, q) \in T_\varepsilon(\lambda)\}$. This number is called the ε-nesting degree of the ε-path λ.

Note that for an arbitrary automaton and any path λ in this automaton, $d_\varepsilon(\lambda) \leq d(\lambda)$, and it is easy to give an example of an automaton and a path λ in this automaton such that $d_\varepsilon(\lambda) < d(\lambda)$.

By $PP_\varepsilon^{(k)}(A)$ ($IP_\varepsilon^{(k)}(A)$) we denote the set of all ε-paths (all initial ε-paths) of the automaton A with ε-nesting degree not exceeding the integer $k \geq 0$.

LEMMA 2 [11]. For an arbitrary paired automaton A, there is a constant $E = E(A, k)$ that depends on A and on the integer k, $k \geq 0$, such that $\forall \lambda \in PP_\varepsilon^{(k)}(A)$, $|\lambda| \leq E$.

COROLLARY. The set $PP_\varepsilon^{(k)}(A)$ for a fixed k is finite.

5. DECIDABILITY OF THE INCLUSION PROBLEM FOR DSUA

Let $A = (Q, \Gamma_1, \Gamma_2, q_0, \delta)$ be an arbitrary paired automaton. Denote by $AP_\varepsilon(A)$ the set of ε-paths in the automaton A that consists of all possible initial ε-paths and their ε-subpaths, i.e., $AP_\varepsilon(A) = \{\mu \in PP_\varepsilon(A) \mid \exists \lambda \in IP_\varepsilon(A)\ (\mu \leq_\varepsilon \lambda)\}$.

Clearly $IP_\varepsilon(A) \subseteq AP_\varepsilon(A) \subseteq PP_\varepsilon(A)$.

Denote by $IQ_\varepsilon(A)$ the set of states reachable by all possible initial ε-paths in the automaton A, i.e., $IQ_\varepsilon(A) = \{q \mid \exists \lambda \in PP_\varepsilon(A, q_0, q)\}$, by $IQ_{\varepsilon\varepsilon}(A)$ the set of states reachable by all possible initial ε-paths λ in the automaton A with $c_2(\lambda) = \varepsilon$, i.e., $IQ_{\varepsilon\varepsilon}(A) = \{q \in IQ_\varepsilon(A) \mid \exists \lambda \in PP_\varepsilon(A, q_0, q)\ (c_2(\lambda) = \varepsilon)\}$.

The automaton A is called ε-periodic on some set of its ε-paths P if for any pair $(u, v) \in T_\varepsilon(A)$ all ε-(u, v)-paths λ from P have the same $c_2(\lambda)$. The automaton A is called ε-periodic if it is ε-periodic on the set $AP_\varepsilon(A)$.

LEMMA 3. An arbitrary paired automaton A is left-stable if and only if it is ε-periodic and $IQ_\varepsilon(A) = IQ_{\varepsilon\varepsilon}(A)$.

Proof. Sufficiency is obvious. Necessity. Let A be left-stable but not ε-periodic. Then the set $AP_\varepsilon(A)$ contains ε-paths λ_1, λ_2 with equal pairs of ends such that $c_2(\lambda_1) \neq c_2(\lambda_2)$. By definition of the set $AP_\varepsilon(A)$, λ_i is a subpath of some initial ε-path μ_i: $\mu_i = \mu_{i1} * \lambda_i * \mu_{i2}$, and $c_2(\mu_i) = \varepsilon$, $i = 1, 2$. Consider the path $\mu_1' = \mu_{11} * \lambda_2 * \mu_{12}$ in the automaton A. In an arbitrary semigroup of words $G = (\Gamma, \Phi)$ with a group system Φ, for any $\alpha_1, \alpha_2 \in \Gamma^*$ we have $\alpha_1 \simeq \alpha_2 \Leftrightarrow (\alpha\alpha_1\beta \simeq \alpha\alpha_2\beta$ for all $\alpha, \beta \in \Gamma^*)$ [5], and therefore $c_2(\mu_1') \simeq c_2(\mu_{11})c_2(\lambda_2)c_2(\mu_{12}) \not\simeq c_2(\mu_{11})c_2(\lambda_1)c_2(\mu_{12}) \simeq c_2(\mu_1)$. But $c_2(\mu_1) = \varepsilon$, and therefore the ε-path μ_1' has $c_2(\mu_1') \neq \varepsilon$, which contradicts stability.

If A is left-stable and $IQ_\varepsilon(A) \neq IQ_{\varepsilon\varepsilon}(A)$, then the automaton A contains initial ε-paths λ such that $c_2(\lambda) \neq \varepsilon$. A contradiction. Q.E.D.

Note that both conditions in the stability criterion of Lemma 3 are essential. Consider the automata $A_i = (Q_i, \Gamma_1, \Gamma_2, q_{0i}, \delta_i)$, $i = 1, 2$, on the alphabets $\Gamma_1 = \{b_1, b_2, b_1^{-1}, b_2^{-1}\}$, $\Gamma_2 = \{c, c^{-1}\}$. $Q_1 = \{q_{01}, q_1, q_2, q_3\}$, $\delta_1 = \{(q_{01}, b_1, c, q_1), (q_1, b_2, c, q_3), (q_1, b_1^{-1}, c^{-1}, q_2), (q_3, b_2^{-1}, \varepsilon, q_1)\}$; $Q_2 = \{q_{02}, q_1\}$, $\delta_2 = \{(q_{02}, \varepsilon, c, q_1)\}$. It is easy to show that in the automaton A_1 we have $IQ_\varepsilon(A_1) = IQ_{\varepsilon\varepsilon}(A_1) = \{q_2\}$ but it is not ε-periodic, while the automaton A_2 is ε-periodic but $IQ_\varepsilon(A_2) \neq IQ_{\varepsilon\varepsilon}(A_2)$, and neither automaton is left-stable.

The following two lemmas establish conditions that are equivalent to the stability conditions of Lemma 3. Let $IQ_\varepsilon^{(0)}(A)$ be the set of states of the automaton A that are reachable by all possible initial ε-paths λ of ε-nesting degree $d_\varepsilon(\lambda) = 0$ and only these states. Similarly $IQ_{\varepsilon\varepsilon}^{(0)}(A)$ denotes the set of states reachable by initial ε-paths λ which have ε-nesting degree $d_\varepsilon(\lambda) = 0$ and $c_2(\lambda) = \varepsilon$.

LEMMA 4. If A is an ε-periodic automaton, then $IQ_{\varepsilon\varepsilon}(A) = IQ_{\varepsilon\varepsilon}^{(0)}(A)$.

Proof. Let A be an arbitrary ε-periodic automaton. Clearly, $IQ_{\varepsilon\varepsilon}^{(0)}(A) \subseteq IQ_{\varepsilon\varepsilon}(A)$. We will show that $IQ_{\varepsilon\varepsilon}(A) \subseteq IQ_{\varepsilon\varepsilon}^{(0)}(A)$. Let $IQ_{\varepsilon\varepsilon}(A) \neq \emptyset$, q an arbitrary state from $IQ_{\varepsilon\varepsilon}(A)$, λ an arbitrary ε-(q_0, q)-path of finite length with $c_2(\lambda) = \varepsilon$. We will show that $q \in IQ_{\varepsilon\varepsilon}^{(0)}(A)$, i.e., there exists an ε-(q_0, q)-path μ with $c_2(\mu) = \varepsilon$ and $d_\varepsilon(\mu) = 0$.

If $d_\varepsilon(\lambda) = 0$, then $\mu = \lambda$ and $q \in IQ_{\varepsilon\varepsilon}^{(0)}(A)$. Let $d_\varepsilon(\lambda) > 0$. From the definition of the ε-nesting degree, there exists a pair of states (u, v) such that $d_\varepsilon(\lambda, u, v) = d_\varepsilon(\lambda) > 0$. Then the ε-path λ contains as ε-subpaths at least one pair of (u, v)-paths μ_1, μ_2 such that $d_\varepsilon(\mu_1, u, v) = 1$, $d_\varepsilon(\mu_2, u, v) = 0$, and $\mu_2 <_\varepsilon \mu_1$. Let $\lambda = \nu_1 * \mu_1 * \nu_2$, where ν_1 and ν_2 are some paths in the alphabet, possibly degenerate. Then the automaton A also contains the path $\lambda_1 = \nu_1 * \mu_2 * \nu_2$. Since λ, μ_1, μ_2 are ε-paths, then λ_1 is an ε-(q_0, q)-path and $c_2(\lambda_1) = c_2(\lambda) = \varepsilon$ by ε-periodicity of the automaton A. Here $|\lambda_1| < |\lambda|$. If $d_\varepsilon(\lambda_1) = 0$, then $\mu = \lambda_1$ and $q \in IQ_{\varepsilon\varepsilon}^{(0)}(A)$. If $d_\varepsilon(\lambda_1) > 0$, then we similarly conclude that there necessarily exists an ε-(q_0, q)-path λ_2 with $c_2(\lambda_2) = \varepsilon$ and $|\lambda_2| < |\lambda_1|$, and so on. Since the path λ is of finite length, there exists $k > 0$ such that the automaton A contains an ε-(q_0, q)-path λ_k with $c_2(\lambda_k) = \varepsilon$ and $d_\varepsilon(\lambda_k) = 0$. Thus, $\mu = \lambda_k$, $q \in IQ_{\varepsilon\varepsilon}^{(0)}(A)$. Q.E.D.

COROLLARY 1. In any paired automaton A, $IQ_\varepsilon(A) = IQ_\varepsilon^{(0)}(A)$.

The proof is similar to the proof of the lemma. From an arbitrary path λ we obtain a path μ with $d_\varepsilon(\mu) = 0$ and, in general, $c_2(\mu) \neq c_2(\lambda)$.

COROLLARY 2. In the ε-periodic automaton A, $IQ_\varepsilon(A) = IQ_{\varepsilon\varepsilon}(A)$ if and only if $IQ_\varepsilon^{(0)}(A) = IQ_{\varepsilon\varepsilon}^{(0)}(A)$.

The proof follows directly from Lemma 4 and Corollary 1.

Let $AP_\varepsilon^{(1)}(A)$ be the set of ε-paths from $AP_\varepsilon(A)$ with ε-nesting degree not higher than 1, i.e., $AP_\varepsilon^{(1)}(A) = \{\lambda \in AP_\varepsilon(A) \mid d_\varepsilon(\lambda) \leq 1\}$. Clearly, $IP_\varepsilon^{(1)}(A) \subseteq AP_\varepsilon^{(1)}(A) \subseteq PP_\varepsilon^{(1)}(A)$.

LEMMA 5. The paired automaton A is ε-periodic if and only if it is ε-periodic on the set $AP_\varepsilon^{(1)}(A)$.

Proof. Necessity is obvious. To prove sufficiency, assume that this is not so: suppose that some automaton A is ε-periodic on $AP_\varepsilon^{(1)}(A)$, but not ε-periodic. Two cases are possible. In the first case, the automaton A contains a pair of states (u, v) such that A is not ε-periodic on the set of ε-(u, v)-paths, and $AP_\varepsilon^{(1)}(A)$ does not contain ε-(u, v)-paths. In the second case, the automaton A contains a pair of states (u, v) such that A is not ε-periodic on the set of ε-(u, v)-paths from $AP_\varepsilon(A)$ but is ε-periodic on ε-(u, v)-paths from $AP_\varepsilon^{(1)}(A)$.

In the first case, there exists a pair of ε-(u, v)-paths μ_1 and μ_2 with $c_2(\mu_1) \neq c_2(\mu_2)$, and μ_1 is of minimum length among all paths μ such that $c_2(\mu) \neq c_2(\mu_2)$. Since $\mu_1 \in AP_\varepsilon(A)$ and $AP_\varepsilon^{(1)}(A)$ does not contain ε-(u, v)-paths, then $d_\varepsilon(\mu_1) > 1$. Then for some pair of states (p, q) the path μ_1 has ε-(p, q)-subpaths ν_1 and ν_2 from $AP_\varepsilon^{(1)}(A)$ such that $\nu_1 <_\varepsilon \nu_2$, and therefore also $|\nu_1| < |\nu_2|$ and $d_\varepsilon(\nu_1) = 0$, $d_\varepsilon(\nu_2) = 1$. Let $\mu_1 = \mu_{11} * \nu_1 * \mu_{12}$, where μ_{11} and μ_{12} are paths. Consider the path $\mu_1' = \mu_{11} * \nu_2 * \mu_{12}$. Clearly μ_1' is an ε-(u, v)-path, and $c_2(\mu_1') = c_2(\mu_1)$, because $c_2(\nu_2) = c_2(\nu_1)$ and, therefore, $|\mu_1'| < |\mu_1|$. A contradiction.

In the second case, there exists an ε-(u, v)-path μ_1 of maximum length among all ε-(u, v)-paths μ from $AP_\varepsilon(A) \backslash AP_\varepsilon^{(1)}(A)$ such that $c_2(\mu) \neq c_2(\lambda)$ for $\lambda \in AP_\varepsilon^{(1)}(A)$. Then $d_\varepsilon(\mu_1) > 1$, and similarly to the first case we obtain a path μ_1' such that $c_2(\mu_1') = c_2(\mu_1)$ and $|\mu_1'| < |\mu_1|$. A contradiction. Q.E.D.

Lemmas 3 and 5 and Corollary 2 of Lemma 4 lead directly to the following lemma.

LEMMA 6. The paired automaton A is left-stable if and only if it is ε-periodic on $AP_\varepsilon^{(1)}(A)$ and $IQ_\varepsilon^{(0)}(A) = IQ_{\varepsilon\varepsilon}^{(0)}(A)$.

Thus, in order to determine if an arbitrary paired automaton A is left-stable, it is necessary and sufficient to determine if it is ε-periodic on the set $AP_\varepsilon^{(1)}(A)$ and if we have $IQ_\varepsilon^{(0)}(A) = IQ_{\varepsilon\varepsilon}^{(0)}(A)$. But by Lemma 2 all the elements of the set $AP_\varepsilon^{(1)}(A)$ are of finite length, while in order to establish equality of the sets $IQ_\varepsilon^{(0)}(A)$ and $IQ_{\varepsilon\varepsilon}^{(0)}(A)$ it suffices to consider the set of paths of finite length $IP_\varepsilon^{(0)}(A)$. Hence we obtain the following theorem.

THEOREM 2. The property of left-stability is decidable on the set of paired automata.

Lemma 1 leads to the following theorem.

THEOREM 3. Inclusion problems for languages K_F-accepted by DSUAs are decidable.

Thus, by Theorems 1 and 3, the set of languages K_F-accepted by DSUAs constitutes a new class of DSULs with a decidable inclusion problem.

LITERATURE CITED

1. P. V. Gorshkov and A. B. Stavrovskii, Finite Automata on the Direct Product of Free Semigroups and Groups [in Russian], Kiev (1988). Unpublished Manuscript, UkrNIINTI 19.10.88, No. 2660-Uk88.
2. E. P. Friedman, "Equivalence problem for deterministic context-free languages and monadic recursive schemas," J. Comput. Syst. Sci., 14, No. 3, 344-359 (1977).
3. S. J. Garlend and D. C. Luckham, Program Schemes, Recursive Schemes and Formal Languages, Rep. ENG-7154, UCLA, Los Angeles (1971).
4. A. Aho and J. Ulmann, The Theory of Parsing, Translation, and Compling [Russian translation], Vol. 1, Mir, Moscow (1978).

5. J. Hopcroft and J. Ulmann, Introduction to Automata Theory, Languages, and Computation, Addison-Wesley, New York—Toronto (1979).
6. J. Hopcroft, "On the equivalence and containment problems for context-free languages," Math. Syst. Theory, 3, No. 1, 119-124 (1969).
7. A. V. Anisimov, "Some subclasses of deterministic context-free languages," Kibernetika, No. 4, 64-67 (1973).
8. S. Greiback and E. Friedman, "Superdeterministic DPDA's: a subcase with a decidable inclusion problem," J. ACM, 27, No. 4, 675-700 (1980).
9. E. S. Lyapin, Semigroups [in Russian], Fizmatgiz, Moscow (1960).
10. A. G. Kurosh, Theory of Semigroups [in Russian], Nauka, Moscow (1967).
11. P. V. Gorshkov and A. B. Stavrovskii, On One Class of DCF Languages with a Decidable Inclusion Problem [in Russian], Kiev (1988). Unpublished Manuscript, UkrNIINTI 17.10.88, No. 2615-Uk88.

ON THE THEORY OF SIGNATURE ANALYZERS

A. S. Barashko

UDC 519.713

An outputless linear sequential machine with zero initial state is used as the mathematical model of a signature analyzer (SA). The main properties of the SA are examined and a criterion of undetectability of an arbitrary set of errors by some SA is derived.

The first report on the use of signature analyzer (SA) for fault detection in digital systems was published more than a decade ago [1]. Signature analyzers are widely used in technical diagnosis, and their properties have been studied by many authors both in the USSR and elsewhere [2-8]. A single binary output sequence is analyzed by a single-channel SA, which is a feedback linear shift register (FLSR) defined by a polynomial whose degree is equal to the number of register bits. For multichannel SAs intended for simultaneous analysis of several binary sequences, an appropriate mathematical model of the SA, according to [4], is a linear sequential machine (LSM) without output [9].

In this paper, we investigate the properties of outputless LSM, some of which are specific only for these machines while others are also observed in LSMs with output. In particular, we establish the necessary and sufficient conditions for representation of an arbitrary set of sequences of binary vectors by some outputless LSM. This set can be interpreted as the set of errors in binary sequences analyzed by a multichannel SA. Knowledge of the general properties of SAs simplifies the analysis of procedures associated with equivalence-preserving transformation of SAs, particularly for the transition from a single-channel SA to its multichannel analog. Note that in some studies this transformation suffers from errors and inaccuracies.

1. EQUIVALENCE AND IRREDUNDANCY OF SIGNATURE ANALYZERS

Definition 1. A signature analyzer is an outputless LSM C = (A, B), where A is a square and B a rectangular binary matrix.

For an integer $p \geq 1$, we denote by E_p the vector space of p-dimensional binary vector columns with the addition operation defined as addition modulo 2. The SA C = (A, B), where A and B are respectively binary (r, r)- and (r, m)-matrices, can be treated as an outputless finite initial automaton $C = (E_r, E_m, \delta, 0)$, where E_r is the set of states, E_m is the input alphabet, δ is the transition function defined by A and B, 0 is the initial state ($0 \in E_r$ and all the components of the vector 0 are zero).

In the multiplication operation for binary matrices, the multiplication of binary elements is defined as conjunction and their addition is defined as addition mod 2 $\oplus$. Then for arbitrary $s \in E_r$ and $x \in E_m$, we define the transition function by the equality $\delta(s, x) = As \oplus Bx$. We extend the function δ in the usual way to the sequences $u \in E_m^*$, where E_m^* is the set of all input sequences of finite length, including the empty sequence e of length 0. To any sequence $u \in E_m^*$ the SA C associates the signature $s(u) = \delta(0, u)$. According to [1], this signature, with an appropriately chosen SA, identifies the sequence u as accurately as fingerprints identify their owner. Errors in the sequence $u \in E_m^*$

Translated from Kibernetika, No. 2, pp. 18-22, March-April, 1990. Original article submitted February 26, 1988.