Analele științifice ale Universității „AL. I. CUZA" din Iași
Tomul XXII, s. I a, 1976, f. 2

DECOMPOSITION OF TIME-VARYING AUTOMATA

BY

DAN A. SIMOVICI

Introduction. *A time-varying automaton* (t. v. a) is a 5-uple $A = (I, S, O, f_t, g_t)$ where I and O are finite sets representing respectively the input and the output alphabet, $S = \{S_1, \ldots, S_t, \ldots\}$ is a denumerable collection of nonempty sets of states subjected to the condition $|\bigcup_{t \in N_1} S_t| < \infty$ (which means that S_t, the set of states of the automaton at time t, is a part of a certain finite set of states $S = \bigcup_{t \in N_1} S_t$). $f_t : S_t \times I \to S_{t+1}$, $g_t : S_t \times I \to 0$ are the next-state and the output functions respectively.

This paper contains an extension of the classical Hartmanis-Stearns theory [3] of decomposition of finite automata to the problem of the decomposition of t. v. a.

In the sequel N will be the set of natural numbers and $N_1 = N \setminus \{O\}$.

The lattice of congruence of a t. v. a. Let us introduce a notion which corresponds to the notion of congruence for automata.

Definition. *A congruence on the t. v. a. $A = (I, S, O, f_t, g_t)$ is a sequence $\{\pi_1, \ldots, \pi_t, \ldots\}$ of equivalences $\{\pi_t \subseteq S_t \times S_t\}$ for which*

$$(s', s'') \in \pi_t \text{ implies } (f_t(s', i),\ f_t(s'', i)) \in \pi_{t+1},\ \forall i \in I.$$

A m, n-congruence on the t. v. a. $A = (I, S, O, f_t, g_t)$ is a sequence $\pi_m, \ldots, \pi_n$ of equivalences, $\pi_t \subseteq S_t \times S_t$ for which $(s', s'') \in \pi_t$ implies $(f_t(s', i), f_t(s'', i)) \in \pi_{t+1}$, $m \leqslant t < n$.

We shall denote by $K(A)$ and $K_n^m(A)$, $m \geqslant 1$ the set of congruences and respectively m, n-congruences of the automaton A. Les us consider the partial ordering on $K(A)$ or $K_n^m(A)$ given by: $\sigma \leqslant \tau$ iff $\sigma_t \leqslant \tau_t$, for all t (respectively, for all t, $m \leqslant t \leqslant n$). It is easy to see that, if $\sigma' = (\sigma'_t)$ and $\sigma'' = (\sigma''_t)$, that $\sigma' \vee \sigma'' = \sup\{\sigma', \sigma''\} = (\sigma'_t \wedge \sigma''_t)$ and $\sigma' \wedge \sigma'' = \inf\{\sigma', \sigma''\} = (\sigma'_t \wedge \sigma''_t)$, where the index t runs on N (and respectively on $\{m, \ldots, n\}$). Here $\sigma'_t \vee \sigma''_t$ and $\sigma'_t \wedge \sigma''_t$ are the supremum and the infimum in Eq_t, the lattice of equivalences on the set S_t.

Let $(L_j, \vee, \wedge)$, $j = 1, 2$ be two complete lattices.

Definition [3]. *A pair-algebra is a subalgebra L of the direct product $(L_1 \times L_2, \vee, \wedge)$ of the lattices L_1 and L_2 for which $(O_1, y) \in L$, $\forall y \in L_2$ and $(x, 1_2) \in L$, $\forall x \in L_1$, O_j and 1_j being the first and the last element of the lattice L_j, $j = 1, 2$.*

The *Hartmanis-Stearns operators of the pair algebra L are the mappings $m : L_1 \to L_2$, $M : L_2 \to L_1$ given by*

$$m(x) = \wedge\{y \mid y \in L_2,\ (x, y) \in L\}, \qquad M(y) = \vee\{x \mid x \in L_1,\ (x, y) \in L\}$$

Theorem 1. [3]. *If L is a pair algebra, then:*

i) $(M(y), y)$ *and* $(x, m(x))$ *are in* L;

ii) $x_1 \geqslant x_2$ *implies* $m(x_1) \geqslant m(x_2)$ and $y_1 \geqslant y_2$ *implies* $M(y_1) \geqslant M(y_2)$;

iii) $m(x_1 \vee x_2) = m(x_1) \wedge m(x_2)$; $M(y_1 \vee y_2) \geqslant M(y_1) \vee M(y_2)$;

iv) $m(x_1 \wedge x_2) \leqslant m(x_1) \wedge m(x_2)$; $M(y_1 \wedge y_2) = M(y_1) \wedge M(y_2)$;

v) $y \geqslant m(x)$ iff $x \leqslant M(y)$, iff $(x, y) \in L$;

vi) $M(m(x)) \geqslant x$, $m(M(y)) \leqslant y$;

vii) $M(m(M(y))) = M(y)$ *and* $m(M(m(x))) = K(x)$;

Let now $h : L_1 \to L_2$ the a lattice homomorphism.

Definition. *A h-congruential element of the lattice L_1 is an element $x \in L$, for which $(x, h(x)) \in L$. Let $C_{L.h}$ be the set of all h-congruential elements.*

In view of the 5^{-th} point of the last theorem x is h-congruential element iff $h(x) \geqslant m(x)$, or $x \leqslant M(h(x))$.

We shall consider a new operator on L_1 namely $M_h : L_1 \to L_1$ given by $M_h(x) = M(h(x))$. Then, x is h-congruential iff $x \leqslant M_h(x)$.

Theorem 2. *If M_h is the operator induced by the homomorphism h, then*

i) $M_h(x_1 \vee x_2) \geqslant M(x^1) \vee M_h(x_2)$,
ii) $M_h(x_1 \wedge x_2) = M_h(x_1) \wedge M_h(x_2)$,
iii) $x \leqslant M_h(x)$ iff $(x, h(x)) \in L$,
iv) $x \leqslant M_h(y)$ iff $(x, h(y)) \in L$,
v) $m(M_h(x)) \leqslant h(x)$.

Proof i) We have $M(x_1 \vee x_2) = M(h(x_1 \vee x_2)) = M(h(x_1) \vee h(x_2)) \geqslant M(h(x_1)) \vee M(h(x_2)) = M_h(x_1) \vee M_h(x_2)$, taking into account the 1-st point of Th. 1 and the fact that h is a homomorphism.

The other points are easy to prove.

Theorem 3. *Let L_1 be a n Artinian lattice (that means a lattice having only finite descending chains), $x \in L_1$ and $y_j = \bigwedge_{k=0}^{j} M_h^k(x)$. For every $x \in L_1$ there exists an integer k so that:* $\max\{ z \mid z \in C_{L,h},\ z \leqslant x \} = y_k$.

Proof. The set $\{ y_1, y_2, \dots \}$ is a descending chain, since $y_{j+1} = \bigwedge_{k=0}^{j+1} M_h^k(x) = (\bigwedge_{k=0}^{j} M_h^k(x)) \wedge (\bigwedge_{k=1}^{j+1} M_h^k(x))$, in view of the idempotency of intersection. Hence we obtain

$$y_{j+1} = y_j \wedge M_h(\bigwedge_{k=0}^{j} M_h^k(x)) = y_j \wedge M_h(y_j) \leqslant y_j.$$

Since L_1 is an Artinian lattice, there exists an integer $k \in N$, so that

$$y_1 > y_2 > \dots > y_k = y_{k+1} = \dots$$

For this index k we have $y_{k+1} = y_k = y_k \wedge M_h(y_k)$ and it is possible to infer that $y_k \leqslant M_h(y_k)$, hence y_k is a h-congruential element. Moreover let $z \in L_1$ be a h-congruential element for which $z \leqslant x$.

Taking into account that z is a h-congruential element, we have the ascending chain $z \leqslant M_h(z) \leqslant M_h^2(z) \leqslant \dots \leqslant M_h^j(z) \leqslant \dots$ Therefore $z \leqslant M_h^j(z) \leqslant M_h^j(x)$ for M_h is a monotonic operator and $z \leqslant x$, hence it follows, $z \leqslant \bigwedge_{j=0}^{k} M_h^j(x) = y_k$. The theorem holds also for finite lattices L_1 since every finite lattice is an Artinian one.

Let $A = (I, S, O, f_t, g_t)$ a t. v. a, $L_1 = \prod_{t=m}^{n} Eq_t$, $L_2 = \prod_{t=m+1}^{n} Eq_t$ and $h : L_1 \to L_2$ the mapping $h(\sigma_m, \dots, \sigma_n) = (\sigma_{m+1}, \dots, \sigma_n)$. It is obvious that h is a lattice homomorphism.

Theorem 4. *The set $K_{m,n} = \{(\sigma, \tau) \mid (\sigma, \tau) \in L_1 \times L_2,\ \sigma = (\sigma_m, \dots \sigma_n),\ \tau = (\tau_{m+1}, \dots, \tau_n),\ (s_1, s_2) \in \sigma_j \Rightarrow (f_j(s_1, i), f_j(s_2, i)) \in \tau_{j+1},\ m \leqslant j \leqslant n-1 \}$ is a pair-algebra; σ is a h-congruential element of L_1 iff it is an m, n-congeruen.*

Proof. Let $\sigma^j = (\sigma_m^j, \dots, \sigma_n^j)$, $\tau^j = (\tau_{m+1}^j, \dots, \sigma_n^j)$ $j = 1, 2$ be two pairs (σ^1, τ^1), $(\sigma^2, \tau^2) \in K$ and let $(s', s'') \in \sigma_t^1 \vee \sigma_t^2$.

There exist $s_1, \dots, s_k$, $s' = s_1$, $(s_1, s_2) \in \sigma_t^1$, $(s_2, s_3) \in \sigma_t^1, \dots, s_k = s''$. Hence for every, $i \in I$, we have $(f_t(s_1, i), f_t(s_2, i)) \in \tau_{t+1}^1$, $(f_{t+1}(s_2, i), f_{t+1}(s_3, i)) \in \tau_{t+1}^2, \dots$, a. s. o. and it follows $(f_t(s', i), f_t(s'', i)) \in \sigma_{t+1}^2 \vee \tau_{s+1}^2$.

We have proved that (σ^1, τ^1), $(\sigma^2, \tau^2) \in K$ implies $(\sigma^1 \vee \sigma^2, \tau_i \vee \tau^2) \in K$. By routine verification it is possible to obtain that $(\sigma^1 \wedge \sigma^2, \tau^1 \wedge \tau^2) \in K$, (ι, τ), $(\sigma, \omega) \in K$ for every $\sigma \in L_1$, $\tau \in L_2$, where $\iota = (\iota_m, \dots, \iota_n)$ and $\omega = (\omega_{m+1}, \dots, \omega_n)$. Here ι and ω_j are the least and the greatest element of the lattice Eq_j.

If σ is a h-congruential element of L_1 we have $(\sigma, h(\sigma)) \in L$ which means that $(s_1, s_2) \in \sigma_j$ implies $(f_t(s_1, i), f_t(s_2, i)) \in h(\sigma_j) = \sigma_{j+1}$, for every $i \in I$, $m \leqslant j \leqslant n-1$.

This comes to the fact that σ is an m, n-congruence. It si easy to see that the converse implication holds also.

Let us consider $L = \prod_{t=1}^{\infty} Eq_t$ and $L' = \prod_{t=2}^{\infty} Eq_t$. The mapping $h : L \to L'$, $h((\sigma_1, \dots, \sigma_t, \dots)) = (\sigma_2, \dots, \sigma_t, \dots)$ is a lattice-homemorphism.

Similarly to the case of m, n-congruences the set $K_\infty = \{ (\sigma, \tau)\ \sigma \in L,\ \tau \in L',\ (s', s'') \in \sigma_t \}$ implies $(f_t(s', i), f_t(s'', i)) \in \sigma_{t+1}\}$ is a pair algebra but the pair algebra $K_{m,n}$ is a finite lattice thus having finite descending chains.

The algorithm described in Th. 3 is effective only for the case of $K_{m,n}$ lattices. This is the reason for which we use the concept of m, n-decomposition rather than the decomposition since in the characterization of m,n-decomposable t. v. a's only m,n-congruences are involved.

Serial, parallel and monitorial decompositions of t. v. a. We shall define here the serial and parallel connection of t. v. a. generalizing for the case of t. v. a. the well known notions from the theory of finite automata. Regarding the monitorial control of a t. v. a., this concept was introduced by A. Salomaa in [4] and it is a specific concept of t. v. a.-theory.

Let $A_1 = (I_1, S_1, O_1, f_{1t}, g_{1t})$, $A_2 = (O_1, S_2, O_2, f_{2t}, g_{2t})$ two t. v. a. the output alphabet of the first automaton being the same as the input alphabet of the second.

Definition. *The serial connection of A_1 and A_2 is the t. v. automaton $A_1 \to A_2 = (I_1, S_1 \times S_2, O_2, f_t, g_t)$, where $S_1 \times S_2 = \{ S_{1t} \times S_{2t}\ t \in N_1,\ S_{jt} \in S_j,\ j = 1, 2 \}$ and*

$$f_t((s_1, s_2), i) = (f_{1t}(s_1, i), f_{2t}(s_2, g_{1t}(s_1, i))), \qquad g_t((s_1, s_2), i) = g_{2t}(s_2, g_{1t}(s_1, i)).$$

The parallel connection of the automata $A_j = (I_j, S_j, O_j, f_{jt}, g_{jt})$ $j = 1, 2$ is the t. v. automaton $A_1 \parallel A_2 = (I_1 \times I_2, S_1 \times S_2, O_1 \times O_2, f_t, g_t)$, where

$$f_t((s_1, s_2), (i_1, i_2)) = (f_{1t}(s_1, i_1), f_{2t}(s_2, i_2)), \qquad g_t((s_1, s_2), (i_1, i_2)) = (g_{1t}(s_1, i_1), g_{2t}(s_2, i_2)).$$

Let now $A = (I, S, O, f_t, g_t)$ be a. t. v. a.

Definition. *A monitor for the t. v. a A is a finite Moore automaton $M = (I, \Sigma, O', \varphi, \gamma)$, where I is the input alphabet, Σ is the set of states, O' is the output alphabet, $O' \subseteq (\bigcup_{t \in N_1} S_{t+1}^{S_t}) \times \bigcup_{t \in N_1} O^{S_t})$, $\varphi : \Sigma \times I \to \Sigma$ is the next-state function and $\varphi : \Sigma \to O'$ is the output function.*

M is a *pure monitor* if φ does not depend effectively on $i \in I$.

The t. v. a A is controlled by M, if at any moment the next state function $f_t : S \to S_{t+1}$ is obtained as an output from the monitor $M : f_t = pr_1 \gamma(s)$, $s \in \Sigma$ and the output function g_t is $g_t = pr_2 \gamma(s)$.

The pair (A, M) can be regarded as a single t. v. .a. by considering the t. v. a. $\hat{A} = (I, \Sigma \times S, O, F_t, G_t)$, where we have used the notation $\Sigma \times S = \{ \Sigma \times S_1, \dots, \Sigma \times S_t, \dots \}$, $F_t((\bar{s}, s), i) = \varphi(\bar{s}, i)$, $pr_1 \gamma(\bar{s})(s, i))$. $G_t((\bar{s}, s), i) = pr_2 \gamma(\bar{s})(s, i)$, $\bar{s} \in \Sigma$, $s \in S_t$, $pr_1 \gamma(\bar{s}) : S_t \to S_{t+1}$, $pr_2 \gamma(s) : S_t \to O$.

Definition. *An m, n-simulation of the t. v. a. $A = (I, S, O, f_t, g_t)$ by the t. v. a $A' = (I', S', O', f_t', g_t')$ is a triple $(h, \{ \alpha_t\ m \leqslant t \leqslant n \}, k)$, where $h : I \to I'$, $\alpha_t : S_t \to S_t'$, $k : O' \to O$, $m \leqslant t \leqslant n$ are mappings for which the diagrams:*

$$\begin{tikzcd} S_t \times I \arrow[r, "g_t"] \arrow[d, "(\alpha_t, h)"'] & O' \\ S_t' \times I' \arrow[r, "g_t'"] & O' \arrow[u, "h"'] \end{tikzcd} \qquad \begin{tikzcd} S_t \times I \arrow[r, "f_t"] \arrow[d, "(\alpha_t, h)"'] & S_{t+1} \arrow[d] \\ S_t' \times I' \arrow[r, "f_t'"] & S_{t+1}' \end{tikzcd}$$

are commutative. This requirement is equivalent to:

(1) $g_t(s, i) = k(g_t'(\alpha_t(s), h(i)))$, (2) $\alpha_{t+1}(f_t(s, i)) = f_t'(\alpha_t(s), h(i))$.

If A is m, n-simulated by A' that means that A' is able to perform at least the computations performed by A using suitable input codifications (h) and output codifications (k) during the period $m \leqslant t \leqslant n$.

$A(1, \infty)$-simulation (or, simply a simulation) is a system $(h, \{\alpha_t \mid t \in N_1\}, k)$ for which the previous diagrams are commutative.

By a (m, n)-serial (parallel, monitoriel) decomposition of a t. v. a. A, we mean a $(m\ n)$-simulation of A by a serial connection (or, respectively a parallel connection or a monitorially contreled t. v. a).

Theorem 5. *Let $A = (I, S, O, f_t, g_t)$ be a t. v. a. There exists a serial (an m, n-serial) decomposition of the t. v. a. A iff there exists a congruence (an m, n-congruence) on A.*

Proof. Let $\sigma = (\sigma_1, \ldots, \sigma_t, \ldots)$ be a congruence on A and consider the t. v. a. $A/\sigma = (I, S^\sigma, O^\sigma, f_t^\sigma, g_t^\sigma)$, where $S^\sigma = (S_1/\sigma_1, \ldots S_t/\sigma_t, \ldots)$. Since a t. v. a has only a finite number of sets of states (for $\bigcup_{t \in N_1} S_t$ is a finite set) and every set of states is a finite one, it follows that S^σ has only a finite number of distinct members. Therefore we can take $O^\sigma = (\bigcup_{t \in N_1} S_t/\sigma_t) \times I$ and $f_t^\sigma([s]_{\sigma_t}, i) = [f_t(s, i)]_{\sigma_{t+1}} \in S_{t+1}/\sigma_{t+1}$, $g_t^\sigma([s]_{\sigma_t}, i) = ([s]_{\sigma_t}, i)$, for every $s \in S_t$, $i \in I$.

We shall construct on every set S_t an equivalence τ_t which is ortogonal on the equivalence $\sigma_t(\sigma_t \wedge \sigma_t = \tau_t)$. This can be performed by numbering the states of each block of the set S_t/σ_t and then placing all the states with the same number in the same block of the set S_t/τ_t. It is obvious that S_t/τ_t has as many elements as the greatest block of the set S_t/σ_t. Now, let us consider the t. v. a. $A^\tau = (O^\sigma, S^\tau, O, f_t^\tau, g_t^\tau)$, where $S^\tau = (S_1/\tau_1, \ldots S_t/\tau_t, \ldots)$,

$$f_t^\tau([s]_{\tau_t}, ([s']_{\tau_t}, i)) = \begin{cases} [f_t([s]_{\tau_t} \cap [s']_{\sigma_t}, i)]_{\tau_{t+1}}, & \text{if } [s]_{\tau_t} \cap [s']_{\sigma_t} \neq \emptyset \\ \text{arbitrary, otherwise,} \end{cases}$$

$$g_t^\tau([s]_{\tau_t}, ([s']_{\tau_t}, i)) = \begin{cases} g_t([s]_{\tau_t} \cap [s']_{\sigma_t}, i), & \text{if } [s]_{\tau_t} \cap [s']_\sigma \neq \emptyset \\ \text{arbitrary, otherwise.} \end{cases}$$

The correctness of the definition of f_t^τ, g_t^τ relies on the fact that arbitrary blocks $[s]_{\tau_t}$, $[s']_{\sigma_t}$ either have a common element or have a void intersection.

Let $\alpha_t : S_t \to S_t/\sigma_t \{S_t/\tau_t\}$ be the mapping given by $\alpha_t(s) = ([s]_t, [s]_{\tau_t})$. We claim that the triple $(1_I, \{\alpha_t \mid 1 \leqslant t < \infty\}, 1_O)$ is a simulation of the t. v. a A by the serial connection $A/\sigma \to A^\tau = (I, S^\tau \times S, O, \overline{f_t}, \overline{g_t})$. We have to check the conditions (1) and (2) We obtain:

$$k(\overline{g_t}(\alpha_t(s), h(i))) = \overline{g_t}(([s]\sigma_t, [s]_{\tau_t}), i) = g_t^\tau([s]_{\tau_t}, g_t^\sigma([s]_{\sigma_t}, i)) = g_t^\tau([s]_{\tau_t}, ([s]_{\sigma_t}, i)) = $$
$$= g_t(s, i), \textit{ and}$$

$$\overline{f_t}(\alpha_t(s), h(i)) = \overline{f_t}(([s]_{\sigma_t}, [s]_{\tau_t}), i) = (f_t^\sigma([s]_{\sigma_t}, i), f_t^\tau([s]_{\tau_t}, ([s]_{\sigma_t}, i))) =$$
$$= ([f_t(s, i)]_{\sigma_{t+1}}, [f_t(s, i)]_{\tau_{t+1}}) = \alpha_{t+1}(f_t(s, i)),$$

hence $(1_I, \{\alpha_j \mid 1 \leqslant t < \infty\}, 1)$ is indeed a simulation. Conversely, let $(h, \{\alpha_t \mid \leqslant t < \infty\}, k)$ be a simulation of the t. v. a $A = (I, S, O, f_t, g_t)$ by the serial connection $A_1 \to A_2$, where $A_1 = (I_1, S_1, O_1, f_{1t}, g_{1t})$, $A_2 = (O_1, S_2, O_2, f_{2t}, g_{2t})$, $A_1 \to A_2 (I_1, S_1, \times S_2, O_2, \overline{f_t}, \overline{g_t})$. The sequence of equivalences $\sigma = (\sigma_1, \ldots, \sigma_t, \ldots)$, where

$$\sigma_t = \{(s', s'') \mid s', s'' \in S_t, \ \mathrm{pr}_1 \alpha_t(s') = \mathrm{pr}_1 \alpha_t(s'')\}$$

is a congruence in A. Indeed, if $(s', s'') \in \sigma_t$, we have $\alpha_t(s') = (s, s_2)$, $\alpha_t(s'') = (s, s_2)$ and therefore $\alpha_{t+1}(f_t(s', i)) = \overline{f_t}(\alpha_t(s_2'), i) = \overline{f_t}((s, s_2), i) = f_{1t}(s, i), f_{2t}(s_2', g_{1t}(s, i))$, $\alpha_{t+1}(f_t(s'', i)) = (f_{1t}(s, i), f_{2t}(s_2'', g_{1t}(s, i))$, which implies $\mathrm{pr}_1 \alpha_{t+1}(f_t(s', i)) = \mathrm{pr}_1 \alpha_{1+t}(f_t(s'', i))$, hence $(f_t, (s', f_t(s'', i)) \in \sigma_{t+1}$. The case of m, n-simulation can be treated analogously.

Theorem 6. *Let $A = (I, S, O, f_t, g_t)$ be a t.v.a. There exists a parallel (an m, n-parallel) decomposition of the t. v. a. A, using an injective simulation (for which α_t are all injective mappings) if there exists two congruences (two m,n-congruences) on A, σ and τ so that $\sigma \wedge \tau = (\iota_{S_1}, \ldots, \iota_{S_t}, \ldots)$ (respectively $\sigma \wedge \tau = (\iota_{S_m}, \ldots, \iota_{S_n})$).*

Proof. Let us consider the t. v. a. $A/\sigma = (I, S^\sigma, O^\sigma, f^\sigma, g^\sigma)$ and $A/\tau = (I^\tau, S^\tau, O_t^\tau, f, g_t^\tau)$ as in the previous theorem. The t. v. a. $A/\sigma \parallel A/\tau = (I \times I, S^\sigma \times S^\tau, O^\sigma \times O^\tau, \overline{f_t}, \overline{g_t})$ is able to simulate A, using the simulation $(h, \{\alpha_t \mid 1 \leqslant t < \infty\}, k)$, where the mappings $h : I \to I \times I$ $\alpha_t : S \to S^\sigma \times S^\tau$, $k : O^\sigma \times O^\tau \to O$ are given by:

$$h(i) = (i, i), \quad \alpha_t(s) = ([s]_{\sigma_t}, [s]_{\tau_t}), \quad k(([s]_{\sigma_t}, i), ([s']_{\tau_t}, i')) =$$
$$= \begin{cases} g_t([s]_{\sigma_t} \cup [s']_{\tau_t}, i), & \text{if } [s]_{\sigma_t} \cup [s']_{\tau_t} \neq \emptyset \textit{ and } i = i' \\ \textit{arbitrary, otherwise.} \end{cases}$$

Since $\sigma_t \wedge \tau_t = \iota_{S_t}$, it follows that every α_t is injective. The conditions (1) and (2) can be verified immediately:

$$k(\overline{g_t}(\alpha_t(s), h(i))) = k(\overline{g_t}(([s]_{\sigma_t}, [s]_{\tau_t}), (i, i))) = k(g_t^\sigma([s]_{\sigma_t}, i), g_t^\tau([s]_{\tau_t}, i)) =$$
$$= k(([s]_{\sigma_t}, i), ([s]_{\tau_t}, i)) = g_t(s, i),$$

and

$$\overline{f_t}(\alpha_t(s), h(i)) = \overline{f_t}(([s]_{\sigma_t}, [s]_{\tau_t}), (i, i)) = ([f_t(s, i)]_{\sigma_{t+1}}, [f_t(s, i)]_{\tau_{t+1}}) = \iota\alpha_{+1}(f_t(s, i)).$$

If there exists a simulation $(h, \{\alpha_t \ 1 \leqslant t < \infty\}, k)$ of the automaton $A = (I, S, O, f_t, g_t)$ by the parallel connection $A_1 \parallel A_2 = \overline{A} = (I_1 \times I_2, S_1 \times S_2, O_1 \times O_2, \overline{f_t}, \overline{g_t})$ of the t. v. a. $A_t = (I_j, S_j, O_j, f_{jt}, g_{jt})$ then we can take the equivalences:

$$\sigma_t = \{(s', s'') \mid s', s'' \in S_t, \mathrm{pr}_1 \alpha_t(s') = \mathrm{pr}_2 \alpha_t(s'')\},$$
$$\tau_t = \{(s', s'') \mid s', s'' \in S_t, \mathrm{pr}_2 \alpha_t(s') = \mathrm{pr}_2 \alpha_t(s'')\},$$

since $\alpha_t : S_t \to S_{1t} \times S_{2t}$.

If $(s', s'') \in \alpha_t$, then $\alpha_t(s') = (s, s')$ and $\alpha_t(s'') = (s, s_1'')$. In view of the properties of simulations we obtain:

$$\alpha_{t+1}(f_t(s', i)) = \overline{f_t}((s, s_1'), (i_1, i_2)) = (f_{1t}(s, i_1), f_{2t}(s_1', i_2))$$

and

$$\alpha_{t+1}(f_t(s'', i)) = (f_{1t}(s, i_1), f_{2t}(s_1'', i_2)).$$

Hence $\mathrm{pr}_1 \alpha_{t+1}(f_t(s', i)) = \mathrm{pr}_1 \alpha_{t+1}(f_t(s'', i))$ and $(f_t(s', i), f_t(s'', i)) \in \alpha_{t+1}$; we have taken $h(i) = (i_1, i_2)$.

The same argument points that $\tau = (\tau_1, \ldots, \tau_t, \ldots)$ is also a congruence.

Afterwards, if $(s', s'') \in \sigma_t \wedge \tau_t$, then $\mathrm{pr}_j \alpha_t(s') = \mathrm{pr}_j \alpha_t(s'')$, $j = 1, 2$, hence $\alpha_t(s') = \alpha_t(s'')$. Taking into account that α_j are injective mappings, we obtain $s' = s''$ which implies $\sigma_t \wedge \tau_t = \iota_t$, $\forall t \in N$.

The case of m, n-decompositions can be treated analogously.

Remark. Using Theorem 3, it is possible to find two congruences σ and τ, with $\sigma \wedge \tau = \iota$ starting from two sequences of equivalences ρ and ρ' for which $\rho \wedge \rho' = \iota$. Specifically, it is possible to take

$$\sigma = \bigwedge_{k=0}^{j} M_h^k(\rho), \qquad \tau = \bigwedge_{k=0}^{j'} M_h^k(\rho'),$$

for sufficiently large j and j'.

Theorem 7. *Let $A = (I, S, O, f_t, g_t)$ be a t. v. a. There exists a monitorial decomposition $(m\ n)$-decomposition of A iff there exists a congruence $\pi = (\pi_1, \ldots, \pi_t, \ldots)$ on A.*

Proof. Let us suppose that A can be simulated by the monitorially controled automaton $A' = (I', S', O', f_t', g_t')$, where the monitor M is $M = (I', \Sigma, O_1, \varphi, \gamma)$. As we have seen above the pair (A', M) can be conceived as the t. v. a. $\hat{A} = (I, \Sigma \times S', O', F_t, G_t)$, where $F_t((\bar{s}, s), i) = (\varphi(\bar{s}, i), \mathrm{pr}_1 \gamma(\bar{s})(s, i))$, $G_t((\bar{s}, s), i) = \mathrm{pr}_2 \gamma(\bar{s})(s, i)$, $\bar{s} \in \Sigma$, $s \in S_t$, $\mathrm{pr}_1 \gamma(\bar{s}) : S_t \to S_{t+1}$, $\mathrm{pr}_2 \gamma(\bar{s}) : S_t \to 0$.

If $(h, \{\alpha_t \ t \in N_1\}, k)$ is the simulation of A by $\hat{A}$, we consider the relation:

$$\sigma_t = \{(s', s'') \ s', s'' \in S_t, \ \mathrm{pr}_1 \alpha_t(s') = \mathrm{pr}_1 \alpha_t(s'')\}.$$

In view of (1) we can write:

$$\alpha_{t+1}(f_t(s, i)) = F_t(\alpha_t(s), h(i)).$$

Applying the definition of π_t we have $\alpha_t(s') = (\bar{s}, s_1')$, $\alpha_t(s'') = (\bar{s}, s_1'')$, if $(s', s'') \in \pi_t$. Thus, we obtain $\alpha_{t+1}(f_t(s', i)) = F_t((\bar{s}, s_1'), i) = (\varphi(\bar{s}, i), \mathrm{pr}_1\, \gamma(\bar{s})(s_1', i))$ and similarly $\alpha_{t+1}(f_t(s'', i)) = (\varphi(\bar{s}, i), \mathrm{pr}_1\, \gamma(\bar{s})(s_1'', i))$. Therefore $\mathrm{pr}_1\, \alpha_{t+1}(f_t(s', i)) = \mathrm{pr}_1\, \alpha_{t+1}(f_t(s'', i))$, hence $(f_t(s', i), f_t(s'', i)) \in \pi_{t+1}$ $\forall i, I$ and π is a congruence.

Conversely, let π be a congruence on the t. v. a $A = (I, S, O, f_t, g_t)$, $\pi = (\pi_1, \ldots, \pi_t, \ldots)$. Using the same technique as in Th. 5 we can construct on every set S_t an equivalence τ_t which is orthogonal on π_t, $\pi_t \wedge \tau_t = \iota_{S_t}$.

We shall use the t. v. a. $A_\tau = (I, S^\tau, O, f_{\tau t}, g_{\tau t})$, where $\tau = (\tau_1, \ldots, \tau_t, \ldots)$, $S^\tau = (S/\tau_1, \ldots, S_t/\tau_t, \ldots)$, the fonctions $f_{\tau t}$ and $g_{\tau t}$ being defined in the sequel. Suppose that the actual state of A_τ is the block $B \in S_t/\tau_t$. Since τ is generally not a congruence, different states of this block are carried in A by a certain symbol i in different blocks of the set S_{t+1}, τ_{t+1}. As monitor, we shall take the Moore automaton $M = (I, \bigcup_{t \in N_1} S_t/\pi_t, O_1, \varphi, \gamma)$, where $\varphi(K, i) = K'$, $K \in S/\pi_t$, $K' \in S_{t+1}/\pi_{t+1}$ and $f_t(K, i) \subseteq K'$.

The correctness of this definition for M follows from the fact that π is a congruence. Since $\pi_t \wedge \tau_t = \iota_{S_t}$ every state $s \in S_t$ is contained in an unique pair of blocks $(K, B) \in S_t/\pi_t \times S_t/\tau_t$. We shall define α_t as $\alpha_t(s) = (K, B)$ and, if $\gamma(K) = (f_{\tau t}, g_{\tau t})$ we take:

$$f_{\tau t}(B, i) = \begin{cases} B', f_t(s, i) \in B' \in S_{t+1}/\tau_{t+1} \text{ and } K \cup B = \{ s \}, \\ \text{arbitrary, if } K \cap B = \emptyset \end{cases}$$

and

$$g_{\tau t}(B, i) = \begin{cases} g_t(s, i), \text{ where } \{ s \} = K \cup SB, \\ \text{arbitrary, if } K \cap B = \emptyset. \end{cases}$$

The t. v. a. related to the pair (A_τ, M) is $\hat{A} = (I, (\bigcup_{t \in N_1} S_t/\pi_t) \times S^\tau, O, F_t, G_t)$, where $F_t((K, B), i) = (\varphi(K, i), \mathrm{pr}_1\, \gamma(K)(B, i))$, $G_t((K, B), i) = \mathrm{pr}_2\, \gamma(K)(B, i)$.

In this special case we can take in (1) and (2) $h = 1_I$ and $k = 1_O$ and through a straightforward calculation we obtain:

$$k(G_t(\alpha_t(s), h(i))) = G_t(\alpha_t(s), i) = G_t((K, B), i) = \mathrm{pr}_2\, \gamma(K)(B, i) = g_{\tau t}(B, i) = g_t(s, i);$$

$$F_t(\alpha_t(s), h(i)) = F_t(\alpha_t(s), i) = F_t((K, B), i) = (\varphi(K, i), \mathrm{pr}_1\, \gamma(K)(B, i)) = (\varphi(K, i), f_{\tau t}(B, i)) = (K', B'),$$

where $f_t(s, i) \in K' \cup B'$, and $\alpha_t(s) = (K, B)$,

which means that $\hat{A}$ is able to simulate A. The same proof (with obvious modifications) works also for m, n-decompositions.

Corollary. *The following three assertions are equivalent for a t. v. a A.*

i) *there exists a congruence on A;*

ii) *there exists a serial decomposition of A;*

iii) *there exists a monitorial decomposition of A.*

These assertions hold also if „congruence" is replaced with „m, n-congruence" and „decomposition" with „m n-decomposition".

Remark. If $(f_t(s, i_1), f_t(s, i_2)) \in \pi_{t+1}$, for every $i_1, i_2 \in I$ the monitor M is a pure one.

REFERENCES

1. Creangă I., Reischer C., Simovici D. — *Introducere algebrică în informatică. Teoria automatelor.* Ed. Junimea, Iași, 1973.
2. Dauscha W., Nürnberg, Starke P. H., Winkler K. D. — *Theorie der determinierten zeitvariablen Automaten.* Electronische Informationsverarbeitung und Kybernetic 9 (1973) 7/8, 455—511.
3. Hartmanis J., Stearns R. E. — *Algebraic Structure Theory of Sequential Machines.* Prentice-Hall, Inc., Englewood Cliffs. 1966.
4. Salomaa A. — *On finite time-variant automata with monitors of different types.* Ann. Univ. Turku, Ser. A, I (1968).

Analele științifice ale Universității „AL. I. CUZA" din Iași
Tomul XXII, s. I a, 1976, f. 2

PERMUTATION AUTOMATA WITH A TIME-VARIANT STRUCTURE

BY

V. FELEA

1. Introduction. Permutation automata have been studied by G. Thierrin [6]. Agasandjan [1, 2] and Arto Salomaa [5] have introduced the notion of automaton with a time-variant structure and solved some of the basic problems concerning time-variant automata.

In this paper the notion of permutation automaton with a time-variant structure is introduced. The automata with a time-variant structure (not only finite) are studied; from this study one obtains the characterization for the finite automata with a time-variant structure and for some families of permutation automata with a time-variant structure. The inclusions between different families of languages accepted by the permutation automata with a time-variant structure are considered. Some operations, under which these families of languages are closed, are determined.

2. Automata with a Time-Variant Structure. Definition 2.1. *A time-variant automaton (shortly t. v. a.) is defined by:*

$$A = < X, S_n, M_n^{n+1}, s_0, F_n, n = 0, 1, 2, \ldots >$$

where:

1) *X is a finite nonempty set of inputs,*

2) *S_n is a finite nonempty set of states at the n-th time instant,*

3) *M_n^{n+1} is a function from $S_n \times X$ into S_{n+1} called the direct function from the n-th time instant into $(n+1)$-th time instant,*

4) *S_0 is the initial state and $s_0 \in S_0$,*

5) *F_n is the set of final states at the moment n, $F_n \subseteq S_n$.* We recursively define the functions M_n^m, with $n \leq m$, by

$$M_n^n(s, e) = e \text{ for every } s \in S_n,\ n = 0, 1, 2, \ldots$$

$$M_n^m(s, p_n^m) = M_{n+1}^m [M_n^{n+1}(s, x_{n+1}),\ p_{n+1}^m]$$

for every $s \in S_n$ and $p_n^m = x_{n+1} \cdots x_m \in X^*$.

Definition 2.2. *Let A be a t. v. a.*

$$T(A) = \{ p_0^n \in X^* / M_0^n(s_0, p_0^n) \in F_n,\ n = 0, 1, 2, \ldots \}$$

is called the set of tapes accepted by the automaton A. A language $L \subseteq X^*$ is accepted by a t. v. a. (shortly X-t. v. a.) iff there exists a t. v. a. A such that $L = T(A)$.

Agasandjan [1, 2] and A. Salomaa [5] define time-variant automata for which $S_n = S$, $n = 0, 1, 2, \ldots$ and S is a finite set. This automaton will be denoted by f. t. v. a.

In the following, we denote the set of the natural numbers by N, $N' = N \cup \{ 0 \}$, the language accepted by an automaton of the type α by αa. The following two definitions appear in [6]:

Definition 2.3. *An equivalence relation R on X^* is said to be right cancellative iff $pq \equiv \bar{q}(R)$ implies that $p \equiv \bar{p}(R)$.*