KYBERNETIKA ČÍSLO 4, ROČNÍK 3/1967

Synthesis of Non-Initial Automata

JÁN ČERNÝ

The paper is concerned with the representation of events by non-initial automata. Two algorithms for the synthesis of such automata starting from the given events $S_1, \ldots, S_n$ are described.

The representation of events by non-initial Medvedyeff automata is defined e.g. in [4] and [5]. In the sequel we shall use the following definition from [5].

Definition 1. Let $\mathbf{A} = (A, X, f)$ be a finite automaton with the set of states A, set of input signals X and transition function f. Let X^* be the set of all strings in X, let $S_1, \ldots, S_n$ be a class of some disjoint events and let $S_i \subset S \subset X^*$, $i = 1, \ldots, n$. Let $N_1, \ldots, N_n$ be the subsets of A. Then we say that $\mathbf{A}$ *represents the events* $S_1, \ldots, S_n$ *on* S *by the sets* $N_1, \ldots, N_n$ if

1. $p \in S_i \Rightarrow f(A, p) \subset N_i$, $i = 1, \ldots, n$, i.e. every string from S_i drives the automaton from an arbitrary state into N_i;

2. $p \in S - S_i \Rightarrow f(A, p) \cap N_i = \emptyset$, $i = 1, \ldots, n$, i.e. every "allowed" string from S, which does not belong to S_i drives the automaton from an arbitrary state into $A - N_i$.

It is evident, that we can assume $S = \bigcup_{i=1}^{n} S_i$. Otherwise we can add the new event $S_{n+1} = S - \bigcup_{i=1}^{n} S_i$ and $\mathbf{A}$ will represent also $S_1, \ldots, S_n, S_{n+1}$. We can also assume that the sets $N_1, \ldots, N_n$ are mutually disjoint.

The problem of non-initial synthesis is the following:

There are given the disjoint events, or, more generally, some events $S_1, \ldots, S_n$ and we have to find the automaton $\mathbf{A} = (A, X, f)$ and the sets $N_i \subset A$, $i = 1, \ldots, n$ such that the non-initial automaton $\mathbf{A}$ will represent the events $S_1, \ldots, S_n$ on $S = \bigcup_{i=1}^{n} S_i$ by $N_1, \ldots, N_n$, or to find out, that such an automaton does not exist. The existence will be strongly connected with the following concept of correct event set.

Definition 2. The set of events $S_1, \ldots, S_n$ is said to be *correct* if $i \leqq n$, $p \in S_i$, $q \in X^*$, $j \neq i \Rightarrow qp \notin S_j$.

It is evident that if $n \neq 1$ and $S_1, \ldots, S_n$ is correct, then $S_1, \ldots, S_n$ is a disjoint set of events not containing the empty string.

The following theorem gives the relation between the two concepts defined above.

Theorem 1. *The set of events $S_i, \ldots, S_n$ in some alphabet X is representable on $S = \bigcup_{i=1}^{n} S_i$ by a non-initial automaton, if and only if it is a correct set.*

Prof. 1. The necessary condition is valid according to the relation

$$f(A, qp) \subset f(A, p) \subset N_i \quad \text{for every} \quad p \in S_i, q \in X^*$$

which implies that $qp \notin S_j \, j \neq i$.

2. In order to prove the sufficient condition let us consider the events $X^*S_1, \ldots$ $\ldots, X^*S_n$ and the *initial* automaton $\mathbf{A}_0 = (A, X, f, a_0)$ which represents them by the sets $N_1, \ldots, N_n \subset A$. We shall assume that $\mathbf{A}_0$ has only the essential states, i.e. that for every state $a \in A$ there exists a string $p_a \in X^*$ such that $f(a_0, p_a) = a$. Such an automaton we can obtain by the well known method of the synthesis described e.g. in [2].

Now let us omit the initial state a_0 and denote $\mathbf{A} = (A, X, f)$. We shall show that $\mathbf{A}$ is the non-initial automaton which represents the events $S_1, \ldots, S_n$ on $S = \bigcup_{i=1}^{n} S_i$. Let $a \in A$ and let p_a be defined as above. Let $p \in S_i$. Then $p_a p \in X^*S_i$ and $f(a, p) = f(a_0, p_a p) \in N_i$. Let $p \in S - S_i$. Then there exists such j that $p \in S_j$ and $f(a, p) = f(a_0, p_a p) \in N_j$. Since $S_1, \ldots, S_n$ was correct $X^*S_1, \ldots, X^*S_n$ are mutually disjoint and therefore the sets $N_1, \ldots, N_n$ are also mutually disjoint [2] and $f(a, p) \notin N_i$. The proof is complete.

Example 1. Let $S_1 = 1(00)^*$, $S_2 = 1(00)^* 0$ (* denotes the iteration), $x = \{0; 1\}$ S_1, S_2 is obviously correct. Let us consider the events

$$R_1 = (1 \vee 0)^* 1(00)^* \quad \text{and} \quad R_2 = (1 \vee 0)^* 1(00)^*0.$$

Let us use the algorithm of synthesis from [2]. Let us denote

$$R_1 = \underset{0}{|}(1\underset{1}{|} \vee 0\underset{2}{|})^* 1\underset{3}{|}(0\underset{4}{|}0\underset{5}{|})^* \qquad R_2 = \underset{0}{|}(1\underset{1}{|} \vee 0\underset{2}{|})^* 1\underset{3}{|}(0\underset{4}{|}0\underset{5}{|})^* 0\underset{6}{|}$$

and after sone restrictions

$$R_1 = \underset{0}{|}(1\underset{0}{|} \vee 0\underset{0}{|})^* 1\underset{1}{|}(0\underset{2}{|}0\underset{1}{|})^* \qquad R_2 = \underset{0}{|}(1\underset{0}{|} \vee 0\underset{0}{|})^* 1\underset{1}{|}(0\underset{2}{|}0\underset{1}{|})^* 0\underset{2}{|}\,.$$

 Now we can already write the table 1 which describes the transition function of the automaton. The corresponding multigraph is shown on the Fig. 1.

The proof of the theorem 1 is constructive, it shows how to design the non-initial automaton, which represents the correct set of events $S_1, \ldots, S_n$. But there remains

Table 1.

	0	$0 \vee 1$	$0 \vee 2$
0	0	$0 \vee 2$	$0 \vee 1$
1	$0 \vee 1$	$0 \vee 1$	$0 \vee 1$

another question: How can we find out that $S_i, \ldots, S_n$ is a correct set. The following theorem gives the answer.

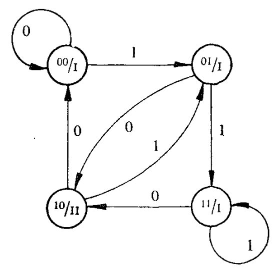

Fig. 1.

Theorem 2. *Let the initial automaton* $\mathbf{A}_0 = (A, X, f, a_0)$ *represent the events* $X^*S_1, \ldots, X^*S_n$ *by the sets* $N_1, \ldots, N_n$. *The set of events* $S_1, \ldots, S_n$ *is correct if and only if* $N_1, \ldots, N_n$ *are mutually disjoint.*

Proof. The necessary condition follows from the proof of the theorem 1. The sufficient condition is implied by the fact that if $S_1, .., S_n$ is not a correct set, then $X^*S_1, \ldots, X^*S_n$ are not mutually disjoint.

Now, according to the theorems 1 and 2 we can write the *algorithm for the non-initial synthesis*:

1. Instead of the given events $S_1, \ldots, S_n$ take the events $X^*S_1, \ldots, X^*S_n$ and find, by the method of [2], the initial automaton $\mathbf{A}_0 = (A, X, f, a_0)$, which represents them by $N_1, \ldots, N_n$ (some subsets of A).

2. Observe, whether $N_1, \ldots, N_n$ are mutually disjoint. If they are not, then $S_1, \ldots$ $\ldots, S_n$ are not representable by any non-initial automaton. If they are disjoint, omit the initial state a_0 and the automaton $\mathbf{A} = (A, X, f)$ is the desired non-initial automaton which represents the events $S_1, \ldots, S_n$ on $S = \bigcup_{i=1}^{n} S_i$ by $N_i, \ldots, N_n$.

The case of ultimate-definite events

An *ultimate-definite event in the alphabet* X is the event of the form X^*S (see [3]). In the sequel we shall consider only such events, for which the event S is finite.

If $R_1, \ldots, R_n$ are such ultimate-definite events, or, more generally, if for every $i = 1, \ldots, n$ the event R_i is a subevent of some event X^*S_i with finite S_i, we need not use the above algorithm if we want to solve the problem of the non-initial synthesis. There is another way (related to [4]):

1. Observe, whether $S_1, \ldots, S_n$ is a correct set of events. Since they are finite the observation will be probably not difficult.

2. Find the legth h of the longest string from $\bigcup_{i=1}^{n} S_i$.

3. Construct the automaton $\mathbf{A} = (A, X, f)$ (with finite memory):

$$A = X^h,$$

$$f((x_1, \ldots, x_h), x) = (x_2, \ldots, x_h, x) \quad \text{for every} \quad (x_1, \ldots, x_h) \in A,\ x \in X$$

and put

$$N_i = \{p \in A: \quad \text{there exist} \quad p_1 \in X^*,\ p_2 \in S_i \quad \text{such that} \quad p_1 p_2 = p\}.$$

In order to show that for every $p \in R_i$ it is $f(A, p) \subset N_i$ we can restrict our consideration only on the strings $p \in S_i$; R_i contains only their elongations to the left.

Choose an arbitrary $a = (x_1, \ldots, x_h) \in A$ and $p \in S_i$. Then

$$f((x_1, \ldots, x_h), p) = \begin{cases} (x_j, \ldots, x_h), p & \text{if the length of } p \text{ is smaller than } h, \\ p & \text{otherwise.} \end{cases}$$

In both cases $f((x_1, \ldots, x_h), p) \in N_i$.

Now let $p \in R_j \neq R_i$. Since $S_1, \ldots, S_n$ was correct, $N_1, \ldots, N_n$ are mutually disjoint and $f(A, p) \cap N_i = \emptyset$. Hence, we can say, that in our special case the last algorithm can replace the previous one.

Example 2.

$$R_1 = (0)^* (1 \vee 00) \subset (1 \vee 0)^* (1 \vee 00),$$

$$R_2 = (1)^* (10) \subset (1 \vee 0)^* (10).$$

We consider the binary alphabet. We have $S_1 = 1 \vee 00$, $S_2 = 10$ and the multigraph of the desired automaton is shown in the Fig. 2. The states from N_i are denoted by I, from N_2 by II.

If $X^*R_1, \ldots, X^*R_n$ are ultimate-definite events and all $R_1, \ldots, R_n$ are finite, then for every $i = 1, \ldots, n$ there exists an event S_i with minimum possible sum of string length, such that $X^*R_1 = X^*S_1, \ldots, X^*R_n = X^*S_n$. Let us take the reverse events

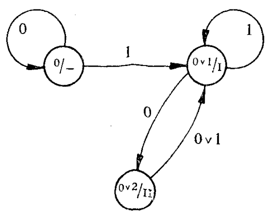

Fig. 2.

$\bar{S}_1, \ldots, \bar{S}_n$ to the events $S_1, \ldots, S_n$ (i.e. $x_1, \ldots, x_n \in \bar{S}_i \Leftrightarrow x_n, \ldots, x_1 \in S_i$) and denote $\bar{S}_i = \{\bar{p}_i(1), \ldots, \bar{p}_i(k_i)\}$, $C = (\bar{p}_1(1), \ldots, \bar{p}_1(k_1), \ldots, \bar{p}_n(k_n))$. Then we see the connection between our considerations and the information theory: $S_1, \ldots, S_n$ is a correct set if and only if the code C enables the unique decoding (see e.g. [1] chapt. 3.1.)

Hence, if the lengths of strings in S_i are denoted $n_i(1), \ldots, n_i(k_i)$, and the number of elements in X is M, then

$$\sum_{i=1}^{n} \sum_{i=1}^{n} M^{-n_i(j)} > 1$$

implies that $S_1, \ldots, S_n$ is not correct (see [1], 3.1).

A problem of J. Hořejš

J. Hořejš stated the following problem: There is a given event R in the alphabet X. What is the maximum $S \supset R$, $S \subset X^*$ such that R is represented on S by a non-initial automaton $\boldsymbol{A} = (A, X, f)$.

It is obvious that S is a solution of the problem, if $T = S - R$ is a maximum event with the property that R, T is a correct set. Let us consider the event $T = X^* - X^*R - \tilde{R}$ where $\tilde{R}$ contains all ends of strings from R (and nothing else). T does not contain any string of the form qp for $p \in R$ and, moreover, if $p \in T$ there is no string $qp \in R$. Hence, R, T is a correct set of events. On the other hand if $T_1 \supset T$, $T_1 \neq T$ then R, T_1 cannot be correct and therefore $T \vee R$ is the maximum event on which R is representable by a non-initial automaton.

(Received December 23th, 1966.)

REFERENCES

[1] Fey P.: Informationstheorie. Berlin 1963.
[2] Глушков В. М.: Синтез цифровых автоматов. Москва 1962.
[3] Paz A., Peleg B.: Ultimate-Definite and Symmetric-Definite events and automata. JACM *12* (1965), 3, 399—410.
[4] Starke P. H.: Über die Darstellbarkeit von Ereignissen in nichtinitialen Automaten. Zeitschr. f. math. Logik u. Grundl. der Math. *9* (1963), 315—319.
[5] Černý J.: O analýze neiniciálnych automatov. Kybernetika *2* (1966), 300—304.

VÝŤAH

Syntéza neiniciálnych automatov

JÁN ČERNÝ

V článku sa uvažujú konečné Medvedovove automaty typu (A, X, f) bez určeného počiatočného stavu (neiniciálne) a definuje sa predstavovanie množiny dizjunktných javov $S_1, \ldots, S_n$ v abecede X na jave S pomocou takéhoto automatu. Popisuje sa algoritmus syntézy, pomocou ktorého je možné ku každej takejto množine javov buď napísať príslušný neiniciálny automat, ktorý ju perdstavuje, alebo zistiť, že taký automat neexistuje. Zvlášť sa uvažujú niektoré typy finálne definitných javov, pro ktoré existuje rýchlejší algoritmus. V závere sa popisuje problém vyhľadania maximálneho javu S, na ktorom je daný jav R predstaviteľný pomocou konečného neiniciálneho automatu.

Dr. Ján Černý, CSc., katedra matematickej analýzy, Univerzita P. J. Šafárika, nám. Febr. víťazstva 9, Košice.