



---

Soviet-era science, translated into English

# Cybernetics and Control Theory

1964

SovietRxiv

---

View the original and related papers at <https://sovietrxiv.org/items/ru-196401.16655>

Source: Math-Net.Ru and CyberLeninka. Machine translation. Verify with the original.

**Abstract**

**Full Text**

## Cybernetics and Control Theory

He Chen-Yu

### On Completely Reversible Turing Machines

*(Presented by Academician P. S. Novikov on 4 VII 1963)*

In the present note the following variant of the definition of a Turing machine is used. We shall consider alphabets

$$a_0, a_1, \dots, a_n; \quad (1)$$

$$q_0, q_1, \dots, q_m, \quad (2)$$

which do not contain the symbols  $L, R, C, |$ . Every word of the form  $q_j a_i | a_k q_l^\theta$ , where  $\theta \in \{L, R, C\}$ , will be called an instruction; the word  $q_j a_i$  is the left-hand side of this instruction, and the word  $a_k q_l^\theta$  is the right-hand side. A Turing machine is a nonempty finite set  $\mathfrak{A}$  of instructions in which no two instructions have graphically equal left-hand sides. The alphabet (1) is called the alphabet of the machine  $\mathfrak{A}$ , and the alphabet (2) the alphabet of states of the machine  $\mathfrak{A}$ . The letter  $a_0$  is called the blank symbol; it will play a special role.  $q_1$  is called the initial state,  $q_0$  the final state of the machine  $\mathfrak{A}$ .

Define the operation  $\underline{\quad}$  on words in the alphabet (1):

$$\underline{A} \rightleftharpoons \begin{cases} a_0, & \text{if } A \in \{a_0\}^*, \\ a_0 a A', & \text{if } A \supseteq A'' a A', \text{ where } A'' \in \{a_0\}, a \neq a_0. \end{cases}$$

Analogously define the operation  $\overline{\quad}$

$$\overline{A} \rightleftharpoons \begin{cases} a_0, & \text{if } A \in \{a_0\}, \\ A' a a_0, & \text{if } A \supseteq A' a A'', \text{ where } A'' \in \{a_0\}, a \neq a_0. \end{cases}$$

A situation of the machine  $\mathfrak{A}$  is any nonlinear expression

$$\underline{A_1} a_i \underline{A_2}, \quad (3)$$

where  $A_1, A_2 \in \{a_0, a_1, \dots, a_n\}$ ,  $a_i \in \{a_0, a_1, \dots, a_n\}$ ,  $q_j \in \{q_0, q_1, \dots, q_m\}$ .

Sometimes, to denote the situation (3), we shall use the notation  $\underline{A_1}^{q_j} \underline{A_2}$ .

The situations  $\underline{A_1}^{q_j} \underline{A_2}$ ,  $\underline{A_3}^{q_l} \underline{A_4}$  of the machine  $\mathfrak{A}$  are called identical if  $\underline{A_1} \supseteq \underline{A_3}$ ,  $\underline{A_2} \supseteq \underline{A_4}$ ,  $r \equiv i$ ,  $l \equiv j$ ; otherwise they are called distinct. If  $j = 1$ , then the situation (3) is called a  $q_1$ -situation. A  $q_1$ -situation of the form  $a_0 \underline{A_2}^{q_1}$  is called initial if either  $a_i \underline{A_2} \overline{\in} \{a_0\}$ , or in the word  $\underline{a_i A_2}$  there are no occurrences of the letter  $a_0$ , not counting the outer letters. If  $j = 0$ , then the situation (3) is called final.

We shall say that, in one step, the machine  $\mathfrak{A}$  transforms the situation (3) (denote it by  $C_1$ ) into a situation  $C_2$ , writing this briefly as  $\mathfrak{A} : C_1 \rightarrow C_2$ , if in the machine  $\mathfrak{A}$  there exists an instruction of the form  $q_j a_i \mid a_k q_l^\theta$  and one of the following three conditions is satisfied:

- 1)  $\theta \supseteq R$  and  $C_2 \supseteq \underline{A_1} a_k \underline{A_2}^{q_j} A'_2$ , where  $a_h A'_2 \supseteq A_2$ ;
- 2)  $\theta \supseteq L$  and  $C_2 \supseteq A'_1 a_h \underline{A_2}^{q_j} a_k A_2$ , where  $A'_1 a_h \supseteq A_1$ ;
- 3)  $\theta \supseteq C$  and  $C_2 \supseteq \underline{A_1} \underline{A_2}^{q_i}$ .

---

\* Following A. A. Markov, we use the notation  $A \overline{\in} \mathcal{A}$  to express that  $A$  is a word in the alphabet  $\mathcal{A}$ .

A situation  $C''$  is called reachable from a situation  $C'$  in the machine  $\mathfrak{A}$  if either  $C'$  and  $C''$  are identical, or there exists a sequence of situations of the machine  $\mathfrak{A}$ :  $C_1, \dots, C_k$  ( $k > 1$ ) such that  $C_1 \equiv C'$ ,  $C_k \equiv C''$ , and for every  $i$  ( $1 \leq i < k$ ) one has  $\mathfrak{A} : C_i \rightarrow C_{i+1}$ . This is denoted by

$$\mathfrak{A} : C' \rightarrow \dots \rightarrow C''.$$

A situation of the machine  $\mathfrak{A}$  is called reachable if it is reachable from some initial situation of the machine  $\mathfrak{A}$ .

A situation  $C$  of the machine  $\mathfrak{A}$  is called elementarily reachable if there exists a situation  $C'$  such that  $\mathfrak{A} : C' \rightarrow C$ . Let  $C_1, C_2, C_3$  be any situations of the machine  $\mathfrak{A}$ ; if the situations  $C_1, C_2$  are not identical and  $\mathfrak{A} : C_1 \rightarrow C_3$ ,  $\mathfrak{A} : C_2 \rightarrow C_3$ , then the situations  $C_1, C_2$  are called merging, each of them nonunique, and  $C_3$  multiple. A state  $q$  of the machine  $\mathfrak{A}$  is called one-sided if in the machine  $\mathfrak{A}$  there are no commands in whose right-hand sides  $q$  would occur with different letters  $\theta$  ( $\theta \in \{L, R, C\}$ ).

A Turing machine is called reversible if it has no merging reachable situations, and completely reversible if it has no merging situations at all.

The question of the existence of a reversible and completely reversible Turing machine equivalent to a given Turing machine was considered by V. S. Chernyavskii<sup>(1)</sup>. He called Turing machines equivalent with respect to an alphabet  $\mathcal{A}$  if, having begun work from one and the same initial situation in the alphabet  $\mathcal{A}$ , they yield one and the same final situation every time that one of them yields some

final situation. V. S. Chernyavskii proved that for every Turing machine  $T$  in an alphabet  $\mathcal{A}$  there exists a reversible Turing machine equivalent to the machine  $T$  with respect to the alphabet  $\mathcal{A}$ , and that for every Turing machine  $T$  there exists a machine equivalent to it with respect to its alphabet which is a composition of three completely reversible machines.

We shall call Turing machines completely equivalent with respect to an alphabet  $\mathcal{A}$  if, having begun work from one and the same  $q_1$ -situation in the alphabet  $\mathcal{A}$ , they yield one and the same final situation every time that one of them yields some final situation.

It is obvious that if machines are completely equivalent with respect to an alphabet  $\mathcal{A}$ , then they are equivalent with respect to this alphabet.

**Theorem 1.** *For every Turing machine  $T$  in an alphabet  $\mathcal{A}$  one can construct a completely reversible machine  $T^{-1}$  such that, for any  $q_1$ -situation  $C_1$  and final situation  $C_0$  of the machine  $T$ , the situation  $C_0$  is reachable from  $C_1$  in the machine  $T$  if and only if the situation  $C_1$  is reachable from  $C_0$  in the machine  $T^{-1}$ .*

**Theorem 2.** *For every Turing machine  $T$  in an alphabet  $\mathcal{A}$  one can construct a completely reversible machine  $T^*$ , completely equivalent to the machine  $T$  with respect to the alphabet  $\mathcal{A}$ .*

Theorem 2 is a strengthening of the above-mentioned results of V. S. Chernyavskii.

The main features of the proof of Theorems 1 and 2 are as follows. For a given machine  $T$  in an alphabet  $\mathcal{A}$  an auxiliary machine  $T_4$  is constructed in this alphabet, completely equivalent to the machine  $T$  with respect to  $\mathcal{A}$  and possessing the following properties: a) all states are one-sided, and no more than two commands can have identical right-hand sides; b) every  $q_1$ -situation is elementarily unreachable, and a  $q_1$ -situation to which the machine  $T_4$  is applicable is nonunique; c) every situation to which the machine  $T_4$  is applicable is either elementarily unreachable, or multiple, or nonunique. From property a) of the machine  $T_4$  it follows that, in the machine  $T_4$ , every situation is reachable from no more than two situations.

For each pair of commands of the machine  $T_4$  with a common right-hand side, let us agree once and for all to call one of them the  $d$ -command and the other the  $b$ -command. Let

$$B \equiv \sigma_1 \cdots \sigma_l, \quad l \geq 0, \quad \sigma_j \in \{b, d\}, \quad i = 1, \dots, l;$$

then the expression

$$T : C_1 \xrightarrow[B]{(5)} \cdots \rightarrow C_2$$

means that in the sequence of situa-

tion  $T : C_1 \rightarrow \dots \rightarrow C_2$  the order of use of the  $\sigma$ -commands, where  $\sigma \doteq b$  or  $\sigma \doteq d$ , is as follows: the  $\sigma_1$ -command, the  $\sigma_2$ -command, ..., the  $\sigma_l$ -command.

Then to each final situation  $C_0$  of the machine  $T_4$  there corresponds a set  $\mathfrak{M}$  of such words  $B$  in the alphabet  $\{b, d\}$  that  $B \in \mathfrak{M}$  if and only if,

( )

there exists a situation  $C$  such that  $T_4 : C \rightarrow \dots \rightarrow C_0$ . The set  $\mathfrak{M}$  is called the **tree** corresponding to the final situation  $C_0$ ; each word of the set  $\mathfrak{M}$  is called a **branch**.

The machine  $T^{-1}$  is constructed from the machine  $T_4$ . Let the alphabet and the alphabet of states of the machine  $T_4$  be, respectively,  $\mathcal{A} = \{a_0, a_1, \dots, a_n\}$  and  $\{q_0, q_1, \dots, q_m\}$ . Then the alphabet of the machine  $T^{-1}$  is

$$\Sigma = \mathcal{A} \cup \{E_i^j\} \cup \{h, d, b, d, \square\}, \quad j = 0, 1, \dots, m; \quad i = 0, 1, \dots, n.$$

We order all words in the alphabet  $\{b, d\}$  as follows: words of different length—in increasing order of length, and words of equal length—lexicographically, regarding the letter  $b$  as preceding the letter  $d$ .

By records of the situation  $A_1 a_i^{q_j} A_2$  of the machine  $T_4$  we shall mean any situations of the machine  $T^{-1}$  of the form

$$[B_1 B_2 \cdot A_1] E_i^j A_2, \quad Q_6$$

where  $Q_6$  is a fixed state of the machine  $T^{-1}$ ,  $B_1 \subseteq \{b, d\}$ ,  $B_2 \subseteq \{b, d\}$ ,  $B_1 B_2 \doteq \Lambda$ , and the word  $[B_1 B_2 \cdot A_1]$  is obtained from the words  $B_1 B_2$  and  $A_1$  in the following way: we replace, from right to left, each occurrence of the letter  $a_0$  in the word  $A_1$  by the letters of the word  $B_1 B_2$ ; moreover, if the length of the word  $B_1 B_2$  is greater than the number of occurrences of the letter  $a_0$  in the word  $A_1$ , then we prefix the remaining letters of the word  $B_1 B_2$  on the left to the already replaced word. The word  $B_1 B_2$  is called the  $\Delta$ -word of this record.

Starting from any final situation  $C_0 \doteq A_3 a_r^{q_0} A_4$  of the machine  $T_4$ , the machine  $T^{-1}$  will operate along the tree  $\mathfrak{M}$  corresponding to the situation  $C_0$  in the following way:

1. It transforms the situation  $A_3 a_r^{q_0} A_4$  into the record of this situation

$$[b, A_3] E_r^0 A_4, \quad Q_6$$

2. Starting from some record  $H$  of an arbitrary situation  $C_2$  of the machine  $T_4$  (in particular,  $C_2$  may be a final situation), the machine will, according to the  $\Delta$ -word  $B_1 B_2$  of the record  $H$ , transform this record in the following

way. If there exists a situation  $C_1$  such that  $T_4 : C_1 \rightarrow C_2$  and  $C_2$  is not multiple, then the machine  $T^{-1}$  transforms this record  $H$  of the situation  $C_2$  into a record of the situation  $C_1$ ; in doing so, the  $\Delta$ -word of the record  $H$  does not change. If the situation  $C_2$  is multiple, i.e. there exist situations  $C_1, C'_1$  such that  $T_4 : C_1 \xrightarrow{(b)} C_2$  and  $T_4 : C'_1 \xrightarrow{(d)} C_2$ , and the  $\Delta$ -word of the record  $H$  has the form  $B_1\sigma B_2$ , where  $\sigma \in \{b, d\}$ , then the machine  $T^{-1}$  transforms this record  $H$  of the situation  $C_2$  either into a record of the situation  $C_1$  or into a record of the situation  $C'_1$ , depending on whether  $\sigma \doteq b$  or  $\sigma \doteq d$ ; in doing so, the letter  $\sigma$  is underlined, and the remaining letters of the word  $B_1\sigma B_2$  do not change. Thus, the machine  $T^{-1}$  will transform any record

$$[B \cdot A_3] E_r^{Q_6} A_4$$

of the final situation  $A_3 a_r A_4$  along the branch  $B$ , converting records into records.

In the process of transforming by the machine  $T^{-1}$  the record

$$[B \cdot A_3] E_r^{Q_6} A_4$$

one of the following three cases may occur: 1) the machine  $T^{-1}$  has not underlined at least one letter of the word  $B$  and has arrived at the record of some elementary unreachable situation of the machine  $T_4$  which is not a  $q_1$ -situation; 2) all letters of the word  $B$  have already been underlined and the machine  $T^{-1}$  has not arrived at the record of a  $q_1$ -situation; 3) the machine  $T^{-1}$  has arrived at the record of a  $q_1$ -situation  $C_1$ . In cases 1) and 2) the machine  $T^{-1}$  returns to the record

$$[B' \cdot A_3] E_r^{Q_6} A_4,$$

where the word  $B'$  immediately follows the word  $B$ . In case 3), if the machine  $T^{-1}$  still-

has not underlined all the letters of the word  $B$ , then it also returns to the record  $[B' \cdot A_3] E_r^{Q_6} A_4$ ; if all the letters of the word  $B$  have already been underlined, then the machine  $T^{-1}$  first transforms this record of the situation  $C_1$  into the situation  $C_1$ , and then returns to the record  $[B' \cdot A_3] E_r^{Q_6} A_4$ .

Thus, starting from any record  $[B \cdot A_3] E_r^{Q_6} A_4$  of the final situation  $A_3 a_r A_4$ , the machine  $T^{-1}$  will sooner or later return to the record  $[B' \cdot A_3] E_r^{Q_6} A_4$  with the  $\Delta$ -word  $B'$  immediately following  $B$ . Next the machine  $T^{-1}$  will, according to the word  $B'$ , transform the record  $[B' \cdot A_3] E_r^{Q_6} A_4$ . Hence, taking 1 into account, it follows that the machine  $T^{-1}$ , starting from the final situation  $A_3 a_r A_4$ , for any  $B \in \{b, d\}$ , transforms the situation  $A_3 a_r A_4$  into the record  $[B \cdot A_3] E_r^{Q_6} A_4$ , and if there exists a  $q_1$ -situation  $C_1$  such that

$$T_4 : C_1 \rightarrow \dots \rightarrow A_3 a_r A_4^{q_0},$$

then the machine  $T^{-1}$  transforms the record  $[B \cdot A_3] E_r^{Q_0} A_4$  into the  $q_1$ -situation  $C_1$ .

If the final situation  $A_3 a_r A_4^{q_0}$  is reachable from some  $q_1$ -situation  $C$ , then from property b) of the machine  $T_4$  it follows that there exists a nonempty word  $B_0 \in \{b, d\}$  such that  $T_4 : C \rightarrow \dots \rightarrow C_0$ . Then the machine  $T^{-1}$  will transform the final situation  $A_3 a_r A_4^{q_0}$  into the record  $[B_0 \cdot A_3] E_r^{Q_0} A_4$ , and then, along the branch  $B_0$ , will seek this  $q_1$ -situation  $C$ . Thus, the machine  $T^{-1}$  can find all  $q_1$ -situations from which the final situation  $A_3 a_r A_4^{q_0}$  is reachable in the machine  $T_4$  (and therefore in the machine  $T$ ).

The machine  $T^*$  indicated in Theorem 2 is easily constructed from the machine  $T^{-1}$ .

Mathematical Institute named after V. A. Steklov  
Academy of Sciences of the USSR

Received  
20 VI 1963

## REFERENCES

1. V. S. Chernyavskii, *Trans. Moscow Math. Soc.*, **9** (1960).

*Note: Figure translations are in progress. See original paper for figures.*

*Source: Math-Net.Ru and CyberLeninka. Machine translation. Verify with the original.*