ELSEVIER Theoretical Computer Science 168 (1996) 215–240

Theoretical Computer Science

Small universal Turing machines

Yurii Rogozhin *

Department of Technical Sciences, Academy of Sciences of Moldova, 1 Stefan cel Mare Avenue, Kishinau, 277612, Moldova

Abstract

Let UTM(m, n) be the class of universal Turing machine with m states and n symbols. Universal Turing machines are proved to exist in the following classes: UTM(24,2), UTM(10,3), UTM(7,4), UTM(5,5), UTM(4,6), UTM(3,10) and UTM(2,18).

1. Introduction

In 1956, Shannon [17] posed the problem of the construction of the simplest universal Turing machine. He was considering ordinary deterministic Turing machine, with a two-way infinite tape and one head. He proposed to measure the complexity of such a Turing machine by the number of commands of this machine, that is the product mn of the number m of states by the number n of tape symbols. It is also possible to consider the number of commands really used by the machine.

Let UTM(m,n) denote the class of universal Turing machines with m states and n symbols. Various definitions of universal Turing machine, and the one we choose, will be discussed in Section 2. Pavlotskaya proved that the classes UTM(3,2) [10] and UTM(2,3) (unpublished) are empty. Using another method, Diekert and Kudlek [3], and Kudlek [5] proved that UTM(2,2) is empty.

The main result of this paper is the following theorem :

Main Theorem. There are universal Turing machines in the following seven classes: UTM(24,2), UTM(10,3), UTM(7,4), UTM(5,5), UTM(4,6), UTM(3,10) and UTM (2,18).

This theorem and the results of Pavlotskaya leave 51 classes UTM(m,n) with an unsettled emptiness problem.

* E-mail: rogozhin@tech.moldova.su.

PII S0304-3975(96)00077-1

Minsky [8] constructed a Turing machine in UTM(7,4), by simulating tag-systems. We use the same method, but the machines in UTM(5,5) and UTM(4,6) are simulating particular classes of tag-systems.

The machines in UTM(24,2), UTM(7,4), UTM(5,5) and UTM(4,6) were already presented in [13]. The machines in UTM(11,3), UTM(3,10) and UTM(2,21) given in [13] are now replaced by machines in UTM(10,3) [14], UTM(3,10) [15] (with 27 commands instead of 28 commands for the one in [13]) and UTM(2,18) [16].

Minsky's machine in UTM(7,4) has an unsolvable halting problem, but has a defect: it damages the output, and, therefore, it cannot compute all partial recursive functions or simulate all Turing machines. Our machine in UTM(7,4) does not have such a defect. Robinson [12] also noticed this defect and gave a machine in UTM(7,4) where the output is preserved and is given immediately to the right of the head of the Turing machine. Robinson considered the number of commands really used in the program of the Turing machine. His machine in UTM(7,4) uses 27 commands, whereas ours uses 26 commands. Note that our Turing machine in UTM(5,5) uses 23 commands, and the one in UTM(4,6) uses 22 commands, which is the least known number of commands for a universal Turing machine. Robinson checked and analysed the machines presented in [13] and the machine in UTM(2,18). Margenstern [7] considers Shannon's problem for non-erasing Turing machines.

The paper is structured as follows. In Section 2, we present various definitions of universality for Turing machines among which two equivalent definitions are retained. The general principles of construction of universal Turing machines simulating tag-systems are presented in Section 3. In Sections 4 to 10, seven universal Turing machines, belonging to the classes mentioned in the *Main Theorem*, are defined and analysed.

2. Equivalent definitions for universal Turing machines

We deal with the ordinary notion of *deterministic Turing machine* (TM) with one-dimensional tape and one head and those of *configuration* of TM and *tag-system* (cf. [2]). First we consider the notion of universality for Turing machines.

In what follows, denote via α, β (with or without subscripts) configurations of a TM (tag-system or other algorithm model) and via $\beta\downarrow$ the fact that the configuration β is final. Let $\beta_1 \overset{M}{\rightarrow} \beta_2$ mean that TM M moves from a configuration β_1 to a configuration β_2 by one step and we write $\beta_1 \overset{M}{\Rightarrow} \beta_j$ for $\beta_1 \overset{M}{\rightarrow} \beta_2 \overset{M}{\rightarrow} \cdots \overset{M}{\rightarrow} \beta_j$.

Let M be a fixed TM. Denote via B the set of all configurations of all Turing machines and via B_M the set of configurations of M. It is well-known that both B and B_M are recursive sets. Also, we define a function F_M as follows:

$$F_M(\alpha) = \beta \quad \text{if and only if} \quad \alpha \overset{M}{\Rightarrow} \beta,$$

where $\alpha, \beta \in B_M$ and $\beta\downarrow$. We denote the domain of F and the range of F with Def(F) and Val(F), respectively.

Definition 1. Let $\psi(n,x)$ be some universal partial recursive function for all partial recursive functions of one variable. A TM U is called universal (UTM), if there exists a total recursive (or simple-recursive) function $\rho(n,x)$, the coding function with $\mathrm{Val}(\rho) \subseteq B_U$, and a recursive function $\lambda(\alpha)$, the decoding function, defined on the set B_U, so that for all n and x the following holds:

$$\lambda(F_U(\rho(n,x))) = \psi(n,x).$$

Remark 1. Definition 1 of the UTM coincides with the definition of the UTM by Davis in [1], except for some immaterial details.

Definition 2. A TM M simulates the TM T, if there exists a recursive function $\tilde{\rho}(\alpha)$, the coding function, defined on the set B_T, with $\mathrm{Val}(\tilde{\rho}) \subseteq B_M$ and a recursive function $\tilde{\lambda}(\alpha)$, the decoding function, defined on the set B_M, with $\mathrm{Val}(\tilde{\lambda}) \subseteq B_T$, so that for all $\alpha \in B_T$, the following holds:

$$\tilde{\lambda}(F_M(\tilde{\rho}(\alpha))) = F_T(\alpha).$$

Remark 2. Definition 2 is based on the definition of the simulation of one abstract computing machine by another according to Herman [4].

Let T_n be a TM with Gödel number n.

Definition 3. A TM U is called a universal Turing machine if one can simulate each TM M and one can effectively get coding and decoding functions from the program of the TM M, i.e. there exists a recursive function $\tilde{\rho}(n,\alpha)$ defined on $N \times B$ with $\mathrm{Val}(\tilde{\rho}) \subseteq B_U$ and a recursive function $\tilde{\lambda}(n,\alpha)$ defined on $N \times B_U$ with $\mathrm{Val}(\tilde{\lambda}) \subseteq B$ so that for all $< n,\alpha > \in N \times B$ the following holds:

$$\tilde{\lambda}(n, F_U(\tilde{\rho}(n,\alpha))) = F_{T_n}(\alpha). \tag{1}$$

Remark 3. It is easy to show that the decoding function can be of one variable, i.e. $\tilde{\lambda}(\alpha)$, $\mathrm{Def}(\tilde{\lambda}) = B_U$, $\mathrm{Val}(\tilde{\lambda}) \subseteq B$ and expression (1) can be rewritten as follows:

$$\tilde{\lambda}(F_U(\tilde{\rho}(n,\alpha))) = F_{T_n}(\alpha). \tag{2}$$

Theorem. *Definitions* 1 *and* 3 *of the universality for Turing machines are equivalent.*

A proof of the *Theorem* above is grounded on the following three lemmas.

Lemma 2.1. *Let M be an UTM with respect to Definition* 1. *Then M calculates arbitrary binary partial recursive function.*

Proof. Obvious. □

Let T_n be the TM with Gödel number n and $G(x)$ be a Gödel enumeration of the set B of all configuration of all TMs.

With each TM M, we associate a number function φ_M as follows:

$$\varphi_M(x) = y \iff x \in G^{-1}(B_M) \& y \in G^{-1}(B_M) \& F_M(G(x)) = G(y). \tag{3}$$

The function $\varphi_M(x)$ is a partial recursive function, because B_M is a recursive set.

Let $F_M(\alpha)$ be not defined, if $\alpha \notin B_M$. It follows from (3) that

$$(\forall x \in N)[\varphi_M(x) = G^{-1} \circ F_M \circ G(x)]. \tag{4}$$

Lemma 2.2. *Each UTM with respect to Definition* 1 *is an UTM with respect to Definition* 3.

Proof. It is obvious that $t(n,x) = \varphi_{T_n}(x)$ is a partial recursive.

According to Lemma 2.1, there are recursive $\lambda(\alpha)$ and $\rho(n,x)$, such that for all $< n,x >$ the following holds:

$$\lambda \circ F_M \circ \rho(n,x) = \varphi_{T_n}(x). \tag{5}$$

From (4) we have

$$(\forall n,x \in N)[\varphi_{T_n}(x) = G^{-1} \circ F_{T_n} \circ G(x)]. \tag{6}$$

Then we have from (5) and (6)

$$\lambda \circ F_M \circ \rho(n,x) = G^{-1} \circ F_{T_n} \circ G(x)$$

and

$$G \circ \lambda \circ F_M \circ \rho(n,x) = F_{T_n} \circ G(x).$$

Let $x = G^{-1}(\alpha)$, where $\alpha \in B$. Then

$$(\forall \alpha \in B)[G \circ \lambda \circ F_M \circ \rho(n, G^{-1}(\alpha)) = F_{T_n}(\alpha)].$$

Let $\tilde{\lambda}(\beta) = G \circ \lambda(\beta)$ for $\beta \in B_M$ and $\tilde{\rho}(n,\alpha) = \rho(n, G^{-1}(\alpha))$, where $\alpha \in B$.

Then $\tilde{\lambda} \circ F_M \circ \tilde{\rho}(n,\alpha) = F_{T_n}(\alpha)$ for all $\alpha \in B$ and TM M is an UTM according to Definition 3. □

Lemma 2.3. *Each UTM according to Definition* 3 *is an UTM according to Definition* 1.

Proof. Let TM M be universal according to Definition 3, then according to Remark 3, there exist recursive functions $\tilde{\lambda}(\alpha)$ and $\tilde{\rho}(n,\alpha)$, such that for all $< n, \alpha > \in N \times B$ the following holds:

$$\tilde{\lambda} \circ F_M \circ \tilde{\rho}(n,\alpha) = F_{T_n}(\alpha).$$

Let T_{n_0} be some UTM according to Definition 1. Then, in particular,

$$\tilde{\lambda} \circ F_M \circ \tilde{\rho}(n_0, \alpha) = F_{T_{n_0}}(\alpha), \tag{7}$$

and according to Definition 1, there exist recursive $\lambda(\alpha)$ and $\rho(n,x)$ such that for all n,x, the following holds:

$$\lambda \circ F_{T_{n_0}} \circ \rho(n,x) = \psi(n,x), \tag{8}$$

where $\psi(n,x)$ is some universal partial recursive function for all unary partial recursive functions.

From (7) and (8),

$$\lambda \circ \tilde{\lambda} \circ F_M \circ \tilde{\rho}(n_0, \rho(n,x)) = \lambda \circ F_{T_{n_0}} \circ \rho(n,x) = \psi(n,x).$$

Let be $\lambda_1(\alpha) = \lambda \circ \tilde{\lambda}(\alpha)$ and $\rho_1(n,x) = \tilde{\rho}(n_0, \rho(n,x))$. Then

$$\lambda_1 \circ F_M \circ \rho_1(n,x) = \psi(n,x). \qquad \square$$

Definition 4 (*Maltsev* [6]). A TM T is called universal, if $\mathrm{Def}(F_T)$ is a creative set.

Remark 4. Maltsev's definition of the universality for TM is, in fact, wider than our two definitions above, because the TM T can have a creative $\mathrm{Def}(F_T)$ and calculate only a constant function.

3. Preliminaries: How to construct a universal Turing machine

The universal Turing machines we define in the following sections simulate tag-systems. For positive integer m and alphabet $A = \{a_1, \ldots, a_n, a_{n+1}\}$, a m-tag-system on A transforms word β on A as follows: we delete the first m letters of β and we append to the right of the result a word that depends on the first letter of β. This process is iterated until m letters cannot be deleted or the first letter is a_{n+1}, and then stops. Formally, we have the following definitions.

Definition 5. A *tag-system* is a three-tuple $T = (m, A, P)$, where m is a positive integer, $A = \{a_1, \ldots, a_{n+1}\}$ is a finite alphabet, and P maps $\{a_1, \ldots, a_n\}$ into the set A^* of finite words (i.e. sequences of letters) on alphabet A and a_{n+1} to *STOP*.

A tag-system $T = (m, A, P)$ is called a m-tag-system. The words $\alpha_i = P(a_i) \in A^*$ are called the *productions* of tag-system T. The letter a_{n+1} is the *halting symbol*. The productions are often displayed as follows:

$$(\mathcal{T}) \quad \begin{cases} a_i \to \alpha_i, & i \in \{1, \ldots, n\} \\ a_{n+1} \to \mathit{STOP} \end{cases}$$

A *computation* of tag-system $T = (m, A, P)$ on word $\beta \in A^*$ is a sequence $\beta = \beta_0, \beta_1, \ldots$ of words on A such that, for all nonnegative integer k, β_k is transformed into

β_{k+1} by deleting the first m letters of β_k and appending word α_i to the result if the first letter of β_k is a_i. The computation stops in k steps if the length of β_k is less than m or the first letter of β_k is a_{n+1}.

Example. The 2-tag-system T_1 is defined by

$$a_1 \to a_2a_1a_3, \quad a_2 \to a_1, \quad a_3 \to STOP.$$

On the initial word $\beta = a_2a_1a_1$, the computation of T_1 is

$$a_2a_1a_1 \to a_1a_1 \to a_2a_1a_3 \to a_3a_1.$$

Minsky [8] proved the existence of a universal 2-tag-system, and, therefore, we will deal only with 2-tag-systems, which also have the following properties:

1. The computation of a tag-system stops only on a word beginning with the halting symbol a_{n+1}.
2. The productions α_i, $i \in \{1,\dots,n\}$, are not empty.

Henceforth, tag-systems will be 2-tag-systems.

A universal Turing machine U simulates a tag-system as follows. Let T be a tag-system on $A = \{a_1,\dots,a_{n+1}\}$ with productions $a_i \to \alpha_i$. To each letter $a_i \in A$ is associated a positive number N_i and codes A_i and $\tilde{A}_i$ (may be $A_i = \tilde{A}_i$), of the form u^{N_i} ($= uu\dots u$, N_i times), where u is a string of symbols of the machine U.

The codes A_i (or $\tilde{A}_i$) are separated by marks on the tape of U.

For $i \in \{1,\dots,n\}$, the production $a_i \to \alpha_i = a_{i1}a_{i2}\dots a_{im_i}$ of the tag-system T is coded by

$$P_i = A_{im_i}A_{im_i-1}\dots A_{i2}A_{i1}.$$

The initial word $\beta = a_ra_sa_t\dots a_w$, to be transformed by the tag-system T, is coded by

$$S = A_rA_sA_t\dots A_w \quad (S = \tilde{A}_r\tilde{A}_s\tilde{A}_t\dots\tilde{A}_w).$$

The initial tape of the UTM is:

$$Q_L\,\underbrace{P_{n+1}P_n\dots P_1P_0}_{P}\,\underbrace{A_rA_sA_t\dots A_w}_{S}\,Q_R,$$

where Q_L and Q_R are respectively infinite to the left and to the right parts of the tape of the UTM and consist only of blank symbols, P_{n+1} is the code of the halting symbol a_{n+1}, P_0 is the additional code consisting of several marks, and the head of the UTM is located on the left side of the code S in the state q_1 (in the case of the machine in UTM(2,18) the head of the UTM is located on the right side of the code P_0 in the state q_1).

Let T be an arbitrary tag-system, S_1 and S_2 be the codes of the words β_1 and β_2, respectively, and $\beta_1 \xrightarrow{T} \beta_2$. Then the UTM U transforms:

$$Q_LP_{n+1}P_n\dots P_1P_0S_1Q_R \overset{U}{\Rightarrow} Q_LP_{n+1}P_n\dots P_1P_0RS_2Q_R$$

(R corresponds to the cells which were bearing the codes of the deleted first two symbols).

The work of the UTM can be divided into three stages:

(i) On the first stage, the UTM searches the code P_r corresponding to the code A_r and then the UTM deletes the codes A_r and A_s (i.e. it deletes the mark between them).

(ii) On the second stage, the UTM writes the code P_r in Q_{R} of the tape in the reversed order.

(iii) On the third stage the UTM restores its own tape for a new cycle of modelling.

The number N_i corresponding to the symbol a_i ($i \in \{1,\dots,n+1\}$) of the tag-system has the property that there are exactly N_r marks at each cycle of modelling between the code P_r and the code A_r (in the case of the machine in UTM(2,18) there are $N_r + 1$ marks, but the additional mark in P_0 is deleted immediately at the beginning of the first stage). On the first stage of modelling, the head of the UTM goes through a number of marks in the part P equal to the number of symbols u in the code A_r.

After the first stage the tape of the UTM is

$$Q_{\mathrm{L}}P_{n+1}P_n \dots P_{r+1}P_rP'_{r-1} \dots P'_1P'_0R'A'_rA_sA_t \dots A_wQ_{\mathrm{R}}$$

and the head of the UTM locates the mark between A'_r and A_s. Then the UTM deletes this mark and the second stage of modelling begins.

After the second stage, the tape of the UTM is

$$Q_{\mathrm{L}}P_{n+1}P_n \dots P_{r+1}P''_rP''_{r-1} \dots P''_1P''_0R''A_t \dots A_wA_{r1}A_{r2} \dots A_{rm_r}Q_{\mathrm{R}},$$

and the head of the UTM is located on the left side of P''_r and the third stage of modelling begins.

After the third stage, the tape of the UTM is

$$Q_{\mathrm{L}}P_{n+1}P_n \dots P_1P_0RA_t \dots A_wA_{r1}A_{r2} \dots A_{rm_r}Q_{\mathrm{R}}$$

and the head of the UTM is located on the right side of R.

Let $a_1, a_2, \dots a_k, b_1, b_2, \dots b_k$ be the symbols of the Turing machine. $a_1a_2 \dots a_kRb_1b_2 \dots b_k$ means that, when the head of the UTM moves to the right the group of symbols $a_1a_2 \dots a_k$ is changed to the group $b_1b_2 \dots b_k$. It is analogous, when the head of the UTM moves to the left (R is changed to L).

$Ra_1a_2 \dots a_k(b_1b_2 \dots b_k)L$ means that the group of symbols $a_1a_2 \dots a_k$ makes the direction of the motion of the head of the UTM change from the right to the left and changes itself to $b_1b_2 \dots b_k$. It is analogous, when the head of UTM moves to the left (R is changed to L).

4. The UTM with 24 states and 2 symbols

The symbols of the machine in UTM(24,2) (see [13]) are 0 (blank symbol) and 1; and the states are q_i ($i = 1,\dots,24$).

$$N_1 = 1, \quad N_{k+1} = N_k + m_k + 2 \quad (k \in \{1,\dots,n\}).$$

The code of the production $\alpha_i = a_{i1}a_{i2}\dots a_{im_i}$ $(i \in \{1,\dots,n\})$ of the tag-system is

$$P_i = 1010(00)^{N_{im_i}}10(00)^{N_{im_i-1}}\dots 10(00)^{N_{i1}}10,$$

where $A_j = (00)^{N_j}$, $j \in \{1,\dots,n+1\}$, and the pair 10 is a mark.

$$P_0 = 10, \qquad P_{n+1} = 1110.$$

The code S of the initial word $\beta = a_r a_s a_t \dots a_w$, to be transformed by the tag-system, is

$$S = (01)^{N_r}11(01)^{N_s}11(01)^{N_t}\dots 11(01)^{N_w},$$

where $\tilde{A}_j = (01)^{N_j}$, $j \in \{1,\dots,n+1\}$, and the pair 11 is a mark.

The program of the machine in UTM(24,2):

$q_1 00Rq_5$	$q_2 01Rq_1$	$q_3 00Lq_4$	$q_4 01Lq_{12}$
$q_1 11Rq_2$	$q_2 11Lq_3$	$q_3 10Lq_2$	$q_4 10Lq_9$
$q_5 01Rq_1$	$q_6 00Lq_7$	$q_7 00Lq_8$	$q_8 00Lq_7$
$q_5 10Lq_6$	$q_6 11Lq_7$	$q_7 10Lq_6$	$q_8 11Rq_2$
$q_9 00Rq_{19}$	$q_{10} 01Lq_4$	$q_{11} 00Lq_4$	$q_{12} 00Rq_{19}$
$q_9 11Lq_4$	$q_{10} 10Rq_{13}$	$q_{11} 1-$	$q_{12} 11Lq_{14}$
$q_{13} 00Rq_{10}$	$q_{14} 00Lq_{15}$	$q_{15} 00Rq_{16}$	$q_{16} 00Rq_{15}$
$q_{13} 11Rq_{24}$	$q_{14} 11Lq_{11}$	$q_{15} 11Rq_{17}$	$q_{16} 11Rq_{10}$
$q_{17} 00Rq_{16}$	$q_{18} 00Rq_{19}$	$q_{19} 01Lq_3$	$q_{20} 01Rq_{18}$
$q_{17} 11Rq_{21}$	$q_{18} 11Rq_{20}$	$q_{19} 11Rq_{18}$	$q_{20} 10Rq_{18}$
$q_{21} 00Rq_{22}$	$q_{22} 01Lq_{10}$	$q_{23} 01Rq_{21}$	$q_{24} 00Rq_{13}$
$q_{21} 11Rq_{23}$	$q_{22} 11Rq_{21}$	$q_{23} 10Rq_{21}$	$q_{24} 10Lq_3$

(i) On the first stage of modelling:

$11L10$	$(q_7 10Lq_6,\ q_6 11Lq_7)$,
$01L00$	$(q_7 10Lq_6,\ q_6 00Lq_7)$,
$00L00$	$(q_7 00Lq_8,\ q_8 00Lq_7)$,
$L10(11)R$	$(q_7 00Lq_8,\ q_8 11Rq_2,\ q_2 01Rq_1)$,
$10R11$	$(q_1 11Rq_2,\ q_2 01Rq_1)$,
$00R01$	$(q_1 00Rq_5,\ q_5 01Rq_1)$,
$R01(00)L$	$(q_1 00Rq_5,\ q_5 10Lq_6, q_6 00Lq_7)$.

If the head of the UTM moves to the right and meets the mark 11, the first stage of modelling is finished.

$R0011(0101)L$ $(q_1 00Rq_5,\ q_5 01Rq_1,\ q_1 11Rq_2, q_2 11Lq_3,\ q_3 10Lq_2,\ q_3 00Lq_4)$ and the second stage of modelling begins.

(ii) On the second stage of modelling:

$01L01$	$(q_4 10Lq_9,\ q_9 00Rq_{19},\ q_{19} 01Lq_3,\ q_3 00Lq_4)$,
$11L10$	$(q_4 10Lq_9,\ q_9 11Lq_4)$,
$0110L0111$	$(q_4 01Lq_{12},\ q_{12} 11Lq_{14},\ q_{14} 11Lq_{11},\ q_{11} 00Lq_4)$.

$L00(01)R$ $(q_4 01Lq_{12},\ q_{12} 00Rq_{19},\ q_{19} 11Rq_{18})$ and the UTM writes the pair 01 in Q_R. In this case:

$10R11$	$(q_{18} 11Rq_{20},\ q_{20} 01Rq_{18})$,
$11R10$	$(q_{18} 11Rq_{20},\ q_{20} 10Rq_{18})$,
$01R01$	$(q_{18} 00Rq_{19},\ q_{19} 11Rq_{18})$,
$R00(01)L$	$(q_{18} 00Rq_{19},\ q_{19} 01Lq_3, q_3 00Lq_4)$.

$L0010(0011)R$ $(q_4 01Lq_{12},\ q_{12} 11Lq_{14},\ q_{14} 00Lq_{15},\ q_{15} 00Rq_{16},\ q_{16} 00Rq_{15},\ q_{15} 11Rq_{17},\ q_{17} 11Rq_{21})$ and the UTM writes the mark 11 in Q_R. In this case:

$10R11$	$(q_{21} 11Rq_{23},\ q_{23} 01Rq_{21})$,
$01R01$	$(q_{21} 00Rq_{22},\ q_{22} 11Rq_{21})$,
$11R10$	$(q_{21} 11Rq_{23},\ q_{23} 10Rq_{21})$,
$R00(11)L$	$(q_{21} 00Rq_{22},\ q_{22} 01Lq_{10},\ q_{10} 01Lq_4)$.

If the head of the UTM moves to the left and meets the group $1110 = P_{n+1}$, then the UTM halts $(q_4 01Lq_{12},\ q_{12} 11Lq_{14},\ q_{14} 11Lq_{11},\ q_{11} 1-)$.

If the head of the UTM moves to the left and meets the group 1010, the second stage of modelling is over:

$L1010(1010)R$ $(q_4 01Lq_{12},\ q_{12} 11Lq_{14},\ q_{14} 00Lq_{15},\ q_{15} 11Rq_{17},\ q_{17} 00Rq_{16},\ q_{16} 11Rq_{10}, q_{10} 10Rq13)$.

(iii) On the third stage of modelling:

$10R10$	$(q_{13} 11Rq_{24},\ q_{24} 00Rq_{13})$,
$01R00$	$(q_{13} 00Rq_{10},\ q_{10} 10Rq_{13})$.

When the head of the UTM moves to the right and meets the mark 11 (in front of the code $\tilde{A}_t$), then both the third stage and the whole cycle of modelling are over. The UTM deletes the mark 11 and a new cycle of modelling begins:

$0111R0101$ $(q_{13} 00Rq_{10},\ q_{10} 10Rq_{13},\ q_{13} 11Rq_{24},\ q_{24} 10Lq_3,\ q_3 10Lq_2,\ q_2 01Rq_1,\ q_1 00Rq_5, q_5 01Rq_1)$.

5. The UTM with 10 states and 3 symbols

The symbols of the machine in UTM(10,3) (see [14]) are 0 (blank symbol), 1 and b; the states are q_i $(i = 1, \ldots, 10)$.

$$N_1 = 2, \quad N_{k+1} = N_k + m_k + t_k \quad (k \in \{1, \ldots, n\}),$$

where if m_k is even, then $t_k = 2$ else $t_k = 1$. Obviously, all N_j ($j \in \{1, \dots n+1\}$) are even.

The code of the production $\alpha_i = a_{i1}a_{i2}\dots a_{im_i}$ ($i \in \{1,\dots,n\}$) of the tag-system, where m_i is odd, is

$$P_i = b0b00^{N_{im_i}}00b00^{N_{im_i-1}}00b\dots 00b00^{N_{i1}}0.$$

If m_i is even, then

$$P_i = b0b0b00^{N_{im_i}}00b00^{N_{im_i-1}}00b\dots 00b00^{N_{i1}}0,$$

where $A_j = 0^{N_j}$, $j \in \{1,\dots,n+1\}$ and the pair $b0$ is a mark.

$$P_0 = b0b0b, \qquad P_{n+1} = 10.$$

The code S of the initial word $\beta = a_r a_s a_t \dots a_w$, to be transformed by the tag-system, is

$$S = 1^{N_r}bb1^{N_s}bb1^{N_t}\dots bb1^{N_w},$$

where $\tilde{A}_j = 1^{N_j}$, $j \in \{1,\dots,n+1\}$ and the pair bb is a mark.

The program of the machine in UTM(10,3):

$q_1 01Rq_1$	$q_2 00Lq_3$	$q_3 00Lq_2$	$q_4 01Rq_1$	$q_5 0bLq_3$
$q_1 10Lq_2$	$q_2 10Lq_2$	$q_3 1bLq_6$	$q_4 11Rq_5$	$q_5 11Rq_5$
$q_1 bbRq_4$	$q_2 bbLq_2$	$q_3 bbRq_1$	$q_4 b1Lq_4$	$q_5 bbRq_5$
$q_6 01Lq_7$	$q_7 00Rq_8$	$q_8 01Lq_6$	$q_9 01Lq_{10}$	$q_{10} 00bRq_5$
$q_6 11Lq_6$	$q_7 1-$	$q_8 11Rq_8$	$q_9 10Rq_{10}$	$q_{10} 10Rq_{10}$
$q_6 bbLq_6$	$q_7 bbLq_9$	$q_8 bbRq_8$	$q_9 b0Lq_4$	$q_{10} bbRq_9$

(i) On the first stage of modelling:

$b1Lb0$	$(q_2 10Lq_2,\ q_2 bbLq_2)$,
$1L0$	$(q_2 10Lq_2)$,
$00L00$	$(q_2 00Lq_3,\ q_3 00Lq_2)$,
$Lb0(b1)R$	$(q_2 00Lq_3,\ q_3 bbRq_1,\ q_1 01Rq_1)$,
$b0Rb1$	$(q_1 bbRq_4,\ q_4 01Rq_1)$,
$0R1$	$(q_1 01Rq_1)$,
$R1(0)L$	$(q_1 10Lq_2)$.

If the head of the UTM moves to the right and meets the mark bb, then the first stage of modelling is finished, and the mark bb is changed to the pair 11:

$Rbb(11)L$ $(q_1 bbRq_4,\ q_4 b1Lq_4)$,

the UTM writes the mark bb in Q_R and the second stage of modelling begins.

(ii) On the second stage of modelling:

$bbLbb$ (q_6bbLq_6),
$b1Lb1$ $(q_611Lq_6,\ q_6bbLq_6)$,
$1L1$ (q_611Lq_6).

$L00(01)R$ $(q_601Lq_7,\ q_700Rq_8,\ q_811Rq_8)$ and the UTM writes the symbol 1 in Q_R. In this case

bRb (q_8bbRq_8),
$1R1$ (q_811Rq_8),
$R0(1)L$ (q_801Lq_6).

$L00b0(01b1)R$ $(q_601Lq_7,\ q_7bbLq_9,\ q_901Lq_{10},\ q_{10}00Rq_5)$ and the UTM writes the mark bb in Q_R. In this case

bRb (q_5bbRq_5),
$1R1$ (q_511Rq_5),
$R10(bb)L$ $(q_511Rq_5,\ q_50bLq_3,\ q_31bLq_6)$.

If the head of the UTM moves to the left and meets the pair $10 = P_{n+1}$, then the UTM halts $(q_601Lq_7,\ q_71-)$.

If the head of the UTM moves to the left and meets the group $b0b0$, then the second stage of modelling is over:

$Lb0b0(b0b0)R$ $(q_601Lq_7,\ q_7bbLq_9,\ q_901Lq_{10},\ q_{10}bbRq_9,\ q_910Rq_{10})$.

(iii) On the third stage of modelling:

$b1Rb0$ $(q_{10}bbRq_9,\ q_910Rq_{10})$,
$1R0$ $(q_{10}10Rq_{10})$.

When the head of the UTM moves to the right and meets the mark bb, both the third stage and the whole cycle of modelling are over. The UTM deletes the mark bb and a new cycle of modelling begins:

$Rbb(10)L$ $(q_{10}bbRq_9,\ q_9b0Lq_4,\ q_4b1Lq_4)$, then in some steps the head of the UTM will be located on the left side of S.

6. The UTM with 7 states and 4 symbols

The symbols of the machine in UTM(7,4) (see [13]) are 0 (blank symbol), 1, b and c; the states are q_i $(i \in \{1,\dots,7\})$.

$$N_1 = 1, \quad N_{k+1} = N_k + m_k + 1 \quad (k \in \{1,\dots,n\}).$$

The code of the production $\alpha_i = a_{i1}a_{i2}\dots a_{im_i}$ $(i \in \{1,\dots,n\})$ of the tag-system is

$$P_i = bb00^{N_{im_i}}b00^{N_{im_i-1}}\dots b00^{N_{i2}}b00^{N_{i1}},$$

where $A_j = 0^{N_j}$, $j \in \{1, \ldots, n+1\}$, and the symbol b is a mark.

$$P_0 = b0, \qquad P_{n+1} = 10.$$

The code S of the initial word $\beta = a_r a_s a_t \ldots a_w$, to be transformed by the tag-system, is

$$S = 1^{N_r} c 1^{N_s} c 1^{N_t} \ldots c 1^{N_w},$$

where $\tilde{A}_j = 1^{N_j}$, $j \in \{1, \ldots, n+1\}$, and the symbol c is a mark.

The program of the machine in UTM(7,4):

$q_1 00Lq_1$	$q_2 01Rq_2$	$q_3 01Lq_4$	$q_4 01Lq_7$
$q_1 10Lq_1$	$q_2 10Lq_1$	$q_3 11Rq_3$	$q_4 11Lq_4$
$q_1 bcRq_2$	$q_2 bcRq_2$	$q_3 bcRq_3$	$q_4 bcLq_4$
$q_1 cbLq_1$	$q_2 c1Rq_5$	$q_3 cbRq_3$	$q_4 cbLq_4$
$q_5 0cLq_4$	$q_6 00Rq_5$	$q_7 00Rq_3$	
$q_5 11Rq_5$	$q_6 10Rq_6$	$q_7 1-$	
$q_5 bcRq_5$	$q_6 bbRq_6$	$q_7 bbLq_6$	
$q_5 cbRq_5$	$q_6 c0Rq_1$	$q_7 c-$	

(i) On the first stage of modelling:

$c1Lb0$	$(q_1 10Lq_1,\ q_1 cbLq_1)$,
$1L0$	$(q_1 10Lq_1)$,
$0L0$	$(q_1 00Lq_1)$,
$b0Rc1$	$(q_2 bcRq_2,\ q_2 01Rq_2)$,
$Lb0(c1)R$	$(q_1 00Lq_1,\ q_1 bcRq_2,\ q_2 01Rq_2)$,
$0R1$	$(q_2 01Rq_2)$,
$R1(0)L$	$(q_2 10Lq_1)$.

If the head of the UTM moves to the right and meets the mark c, then the first stage of modelling is finished, and the mark c is changed to the symbol 1:

$cR1$ $(q_2 c1Rq_5)$,

and the UTM writes the mark c in Q_R and the second stage of modelling begins.

(ii) On the second stage of modelling:

cLb	$(q_4 cbLq_4)$,
bLc	$(q_4 bcLq_4)$,
$1L1$	$(q_4 11Lq_4)$.

$L00(01)R$ (q_401Lq_7, q_700Rq_3, q_311Rq_3) and the UTM writes the symbol 1 in Q_R. In this case:

cRb	(q_3cbRq_3),
bRc	(q_3bcRq_3),
$1R1$	(q_311Rq_3),
$R0(1)L$	(q_301Lq_4).

$L0b0(0c1)R$ (q_401Lq_7, q_7bbLq_6, q_600Rq_5, q_5bcRq_5, q_511Rq_5) and the UTM writes the mark c in Q_R. In this case

cRb	(q_5cbRq_5),
bRc	(q_5bcq_5),
$1R1$	(q_511Rq_5),
$R0(c)L$	(q_50cLq_4).

If the head of the UTM moves to the left and meets the pair $10 = P_{n+1}$, then the UTM halts (q_401Lq_7, q_71-).

If the head of the UTM moves to the left and meets the group $bb0$, then the second stage of modelling is over:

$Lbb0(bb0)R$ (q_401Lq_7, q_7bbLq_6, q_6bbRq_6, q_610Rq_6).

(iii) On the third stage of modelling:

$b1Rb0$	$(q_6bbRq_6,\ q_610Rq_6)$,
$1R0$	(q_610Rq_6).

When the head of the UTM moves to the right and meets the mark c, both the third stage and the whole cycle of modelling are over. The UTM deletes the mark c and a new cycle of modelling begins:

$cR0$ (q_6c0Rq_1).

7. The UTM with 5 states and 5 symbols

The machine in UTM(5,5) (see [13]) simulates the following class of tag-systems:

$$(T_1)\quad \begin{cases} a_i \to b\alpha_i a, & i \in \{1,\dots,n\} \\ a \to \Lambda \\ b \to \Lambda \\ a_{n+1} \to STOP \end{cases}$$

where $\alpha_i = a_{i1}a_{i2}\dots a_{im_i}$ is a finite word in the alphabet $A = \{a_k\}$, $k \in \{1,\dots,n+1\}$ (α_i is not empty) and Λ is the empty word.

We show the universality of the tag-systems of type T_1 in Lemma 7.1.

Lemma 7.1. For every tag-system T of type $\mathcal{T}$ (see Definition 5), there is a tag-system T' of type $\mathcal{T}_1$ which models T.

Proof. We change a fixed tag-system T by adding two new letters a and b with

$$\begin{cases} a_i \rightarrow b\alpha_i a, & i \in \{1,\dots,n\} \\ a \rightarrow \Lambda \\ b \rightarrow \Lambda \\ a_{n+1} \rightarrow STOP \end{cases}$$

We show that if $\beta \overset{T}{\Rightarrow} \gamma$, then $\beta a \overset{T'}{\Rightarrow} \gamma' a$ (the first letter in the word γ' is different from a and b) and γ results from γ' by deleting any occurrences of the symbols a and b.

Induction basis. Let $a_i a_j \beta \overset{T}{\rightarrow} \beta\alpha_i$. Then $a_i a_j \beta a \overset{T'}{\rightarrow} \beta a b \alpha_i a$. If β is empty, then $a_i a_j a \overset{T'}{\rightarrow} ab\alpha_i a \overset{T'}{\rightarrow} \alpha_i a$. Because $\alpha_i \neq \Lambda$, the basis is proved.

Induction hypothesis. Let $\beta \overset{T}{\Rightarrow} \gamma_t \overset{T}{\rightarrow} \gamma$ and $\beta a \overset{T'}{\Rightarrow} \gamma_t' a$, where the first letter in the word γ_t' is different from a and b, and γ_t results from γ_t' by deleting any occurrences of a and b.

We note that in the process of transforming of word βa by the tag-system in the sequence $\beta a \overset{T'}{\rightarrow} \beta_1 a \overset{T'}{\rightarrow} \cdots \overset{T'}{\rightarrow} \beta_j a \overset{T'}{\rightarrow} \cdots$ the words that do not begin with the symbols a and b have the form: $\beta_j = \beta_{j1} ab \beta_{j2} ab \dots \beta_{jk_j - 1} ab \beta_{jk_j}$ (β_{jr}, $r \in \{1, \dots k_j\}$, the symbols a and b do not occur, and that the words β_{j1} and β_{jk_j} are not empty).

Consider two cases:

(i) $\gamma_t' = a_i a_j \delta_t'$, $i,j \in \{1,2,\dots,n+1\}$, $i \neq n+1$. Then $\gamma_t = a_i a_j \delta_t$, where δ_t results from δ_t' by deleting any occurrences of the symbols a and b, and $a_i a_j \delta_t \overset{T}{\rightarrow} \delta_t \alpha_i$ $(\gamma = \delta_t \alpha_i)$, though $a_i a_j \delta_t' a \overset{T'}{\rightarrow} \delta_t' a b \alpha_i a$.

If δ_t is empty, then we take into account that δ_t' is also empty. Then $\delta_t' a b \alpha_i a = ab\alpha_i a \overset{T'}{\rightarrow} \alpha_i a$. Let $\gamma' = \alpha_i$. This is a desirable γ'.

If δ_t is not empty, then we take into account that either the first letter in the word δ_t' is different from a and b (then $\gamma' = \delta_t' a b \alpha_i$) or $\delta_t' = ab\delta_t''$, where the first letter in the word δ_t'' is different from a and b, and δ_t results from δ_t'' by deleting all occurrences of a and b. In the latter case $ab\delta_t'' ab\alpha_i a \overset{T'}{\rightarrow} \delta_t'' ab \alpha_i a$. Let $\gamma' = \delta_t'' ab\alpha_i$. This is a desirable γ'.

(ii) $\gamma_t' = a_i a \delta_t'$, $i \in \{1,2,\dots,n\}$. Then $\delta_t' = b\delta_t''$ and $\gamma_t = a_i \delta_t$, where δ_t results from δ_t'' by deleting any occurrences of a and b.

Let be $\delta_t = a_j \delta_{t1}$, $j \in \{1,2,\dots,n+1\}$. It means $a_i a_j \delta_{t1} \overset{T}{\rightarrow} \delta_{t1}\alpha_i$ $(\gamma = \delta_{t1}\alpha_i)$.

Taking into account that $\delta_t'' = a_j \delta_{t1}'$ and δ_{t1} results from δ_{t1}' by deleting any occurrences of a and b, we have

$$a_i a b a_j \delta_{t1}' a \overset{T'}{\rightarrow} b a_j \delta_{t1}' a b \alpha_i a \overset{T'}{\rightarrow} \delta_{t1}' a b \alpha_i a.$$

If δ_{t1} is empty, then δ'_{t1} is empty as well. Then $\delta'_{t1}ab\alpha_i a \xrightarrow{T'} \alpha_i a$ and $\gamma' = \alpha_i$. If δ_{t1} is not empty, then either δ'_{t1} begins with a letter other than a and b (then $\gamma' = \delta'_{t1}ab\alpha_i$), or $\delta'_{t1} = ab\delta''_{t1}$, where the word δ''_{t1} begins with a letter other than a and b, and δ_{t1} results from δ''_{t1} by deleting any occurrences of a and b. In the latter case, $ab\delta''_{t1}ab\alpha_i a \xrightarrow{T'} \delta''_{t1}ab\alpha_i a$. Let $\gamma' = \delta''_{t1}ab\alpha_i$. This is a desirable γ'. □

The symbols of the machine in UTM(5,5) are 0, 1, b (blank symbol), c and d; and the states are q_i $(i = 1,\ldots,5)$.

$$N_1 = 3, \quad N_{k+1} = N_k + m_k + 4 \quad (k \in \{1,\ldots,n\}).$$

$$N_a = N_{n+1} + 2 \text{ and an arbitrary number } N > N_a,$$

$$N_b = 1.$$

The code of the production $\alpha_i = ba_{i1}a_{i2}\ldots a_{im_i}a$ $(i \in \{1,\ldots,n\})$ of the tag-system is

$$P_i = bb1^{N_a}1b1^{N_{im_i}}1b11^{N_{im_i-1}}b\ldots 1b11^{N_{i1}}1b11^{N_b}1b,$$

where $A_j = 1^{N_j}$, $j \in \{1,\ldots,n+1\}$, $A = 1^{N_a}$, $B = 1^{N_b}$ and the symbol b is a mark.

$$P_0 = bbb, \qquad P_{n+1} = 1b1b.$$

The code S of the initial word $\beta = a_r a_s a_t a \ldots a_w$, to be transformed by the tag-system, is

$$S = 1^{N_r}c1^{N_s}c1^{N_t}\ldots c1^{N_w}c1^{N_a},$$

where $A_j = 1^{N_j}$, $j \in \{1,\ldots,n+1\}$, $A = 1^{N_a}$, $B = 1^{N_b}$ and the symbol c is a mark.

The program of the machine in UTM(5,5):

$q_1 01Rq_1$	$q_2 00Rq_2$	$q_3 0cLq_4$	$q_4 01Lq_4$	$q_5 0-$
$q_1 10Lq_1$	$q_2 10Rq_2$	$q_3 10Rq_3$	$q_4 10Rq_2$	$q_5 11Rq_5$
$q_1 bdRq_1$	$q_2 b0Lq_4$	$q_3 bbRq_5$	$q_4 bdLq_3$	$q_5 b-$
$q_1 c0Rq_2$	$q_2 ccRq_2$	$q_3 ccRq_3$	$q_4 ccLq_4$	$q_5 c1Rq_1$
$q_1 dbLq_1$	$q_2 ddRq_2$	$q_3 ddRq_3$	$q_4 ddLq_4$	$q_5 dbRq_5$

(i) On the first stage of modelling:

dLb	$(q_1 dbLq_1)$,
$1L0$	$(q_1 10Lq_1)$,
$Lb(d)R$	$(q_1 bdRq_1)$,
$0R1$	$(q_1 01Rq_1)$,
bRd	$(q_1 bdRq_1)$,
$R1(0)L$	$(q_1 10Lq_1)$.

If the head of the UTM moves to the right and meets the mark c, the first stage of modelling is finished. Then the mark c is changed to the symbol 0:

$cR0 \quad (q_1c0Rq_2)$,

and the second stage of modelling begins.

(ii) On the second stage of modelling:

dLb	(q_4ddLq_4),
cLc	(q_4ccLq_4),
$0L1$	(q_401Lq_4).

$L1(0)R$ (q_410Rq_2) and the UTM writes the symbol 0 in Q_R. In this case

dRd	(q_2ddRq_2),
cRc	(q_2ccRq_2),
$1R0$	(q_210Rq_2),
$(0R0$	(q_200Rq_2) if the UTM wrote the symbol 0 before that),

then $Rb(0)L$ (q_2b0Lq_4).

As after the first stage of modelling P'_k $(k = 0, \ldots, r-1)$ differs from P_k by the fact that the marks b are changed to the marks d, the UTM will write a number of the symbols 0 in Q_R as many as the number of the symbols 1 between the codes P_r and S. As a result there will be a new code A of the symbol a.

$L1b(0d)R$ $(q_4bdLq_3,\ q_310Rq_3,\ q_3ddRq_3)$ and the UTM writes the mark c in Q_R. In this case

dRb	(q_3ddRq_3),
cRc	(q_3ccRq_3),
$1R0$	(q_310Rq_3),
$R0(c)L$	(q_30cLq_4).

If the head of the UTM moves to the left and meets the group $1b1b = P_{n+1}$, then the UTM will try to write two marks c in Q_R and halts $(q_3bbRq_5,\ q_5b-)$.

If the head of the UTM moves to the left and meets the pair bb, the second stage of modelling is over:

$Lbb(bb)R$ $(q_4bdLq_3,\ q_3bbRq_5,\ q_5dbRq_5)$.

If, at the beginning of a new cycle of modelling $A_r = A$, then on the first stage of modelling the head of the UTM goes to Q_L without a failure and on the second stage of modelling the UTM will write the number of 0 in Q_R as many times as the number of 1s that are in P. As a result there will be a new code A of the symbol a. Then, on the second stage the UTM meets the pair bb in Q_L without a failure and the third stage of modelling begins.

If at the beginning of a new cycle of modelling $A_r = B$, then on the first stage of modelling the UTM meets the pair bb in P_0 and turns immediately to the third stage of modelling.

(iii) On the third stage of modelling:

$dRb \quad (q_5dbRq_5),$
$1R1 \quad (q_511Rq_5).$

When the head of the UTM moves to the right and meets the mark c, both the third stage and the whole cycle of modelling are over. The UTM deletes the mark c and a new cycle of modelling begins:

$cR1 \quad (q_5c1Rq_1).$

8. The UTM with 4 states and 6 symbols

The machine in UTM(4,6) (see [13]) simulates the following class of tag-systems:

$$(\mathcal{T}_2) \quad \begin{cases} a_i \longrightarrow \alpha_i, & i \in \{1,\dots,n\} \\ a_n \longrightarrow a_n a_n \\ a_{n+1} \longrightarrow STOP \end{cases}$$

where $\alpha_i = a_n a_n \beta_i$ and β_i are not empty.

We show the universality of tag-systems of type $\mathcal{T}_2$ in Lemma 8.1.

Lemma 8.1. For every tag-system T of type $\mathcal{T}$, there is the tag-system T' of type $\mathcal{T}_2$ which models T.

Proof. Can be provided in the same manner as that of Lemma 7.1. □

The symbols of the machine in UTM(4,6) are 0 (blank symbol), 1, b, $\overleftarrow{b}$, $\vec{b}$ and c; the states are q_i ($i = 1,\dots,4$).

$$N_1 = 1, \quad N_{k+1} = N_k + 2m_k \quad (k \in \{1,\dots,n\}).$$

We note that $m_n = 2$. The code of the production $\alpha_i = a_n a_n a_{i3} \dots a_{im_i}$ ($i \in \{1,\dots,n\}$) of the tag-system is

$$P_i = b1b1^{N_{im_i}}bb1^{N_{im_i-1}}\dots bb1^{N_{i3}}bb1^{N_n}bb1^{N_n-N_i}.$$

In particular,

$$P_n = b1b1^{N_n}bb, \quad P_0 = b, \quad P_{n+1} = \overleftarrow{b}b,$$

where $A_j = 1^{N_j}$, $j \in \{1,\dots,n+1\}$ and the symbol b is a mark.

The code S of the initial word $\beta = a_r a_s a_t \dots a_w$, to be transformed by the tag-system, is

$$S = 1^{N_r}c1^{N_s}c1^{N_t}\dots c1^{N_w}$$

and the symbol c is a mark.

The program of the machine in UTM(4,6):

$q_1\ 1\overleftarrow{b}\ Lq_1$	$q_2\ 10\ \ Rq_2$	$q_3\ 11\ \ Rq_3$	$q_4\ 10\ \ Rq_4$
$q_1\ b\vec{b}\ Rq_1$	$q_2\ b\vec{b}\ \ Lq_3$	$q_3\ b\overleftarrow{b}\ Rq_4$	$q_4\ bc\ \ Lq_2$
$q_1\ \vec{b}b\ Lq_1$	$q_2\ \vec{b}\overleftarrow{b}\ Rq_2$	$q_3\ \vec{b}b\ Rq_3$	$q_4\ \vec{b}\overleftarrow{b}\ Rq_4$
$q_1\ \overleftarrow{b}0\ Rq_1$	$q_2\ \overleftarrow{b}\vec{b}\ Lq_2$	$q_3\ \overleftarrow{b}\ \ —$	$q_4\ \overleftarrow{b}\ \ —$
$q_1\ 0\overleftarrow{b}\ Lq_1$	$q_2\ 01\ \ Lq_2$	$q_3\ 0c\ \ Rq_1$	$q_4\ 0c\ \ Lq_2$
$q_1\ c0\ \ Rq_4$	$q_2\ cb\ \ Rq_2$	$q_3\ c1\ \ Rq_1$	$q_4\ cb\ \ Rq_4$

(i) On the first stage of modelling:

$1\ L\ \overleftarrow{b}$ $\quad(q_1\ 1\overleftarrow{b}\ Lq_1)$,
$0\ L\ \overleftarrow{b}$ $\quad(q_1\ 0\overleftarrow{b}\ Lq_1)$,
$\vec{b}\ L\ b$ $\quad(q_1\ \vec{b}b\ Lq_1)$,
$L\ b\ (\vec{b})R$ $\quad(q_1\ b\vec{b}\ Rq_1)$,
$\overleftarrow{b}\ R\ 0$ $\quad(q_1\ \overleftarrow{b}0\ Rq_1)$,
$b\ R\ \vec{b}$ $\quad(q_1\ b\vec{b}\ Rq_1)$,
$R\ 1\ (\overleftarrow{b})L$ $\quad(q_1\ 1\overleftarrow{b}\ Lq_1)$.

If the head of the UTM moves to the right and meets the mark c, the first stage of modelling is finished. At that time the tape of the UTM is

$$Q_{\mathrm{L}}P_{n+1}P_n \ldots P_{r+1}P_rP'_{r-1} \ldots P'_1P'_0R'A'_rA_sA_t \ldots A_wQ_{\mathrm{R}}$$

and the head of the UTM locates the symbol c between the codes of A'_r and A_s ($A'_j = 0^{N_j},\ j \in \{1,\ldots,n+1\}$). $P'_k(k \in \{0,\ldots,r-1\})$ differs from P_k by the fact that the marks b are changed to the marks $\vec{b}$ and A_j to A'_j ($j \in \{1,\ldots,n+1\}$):

$$P'_k = \vec{b}0\vec{b}0^{N_{km_k}}\vec{b}\vec{b}0^{N_{km_k-1}} \ldots \vec{b}\vec{b}0^{N_{k3}}\vec{b}\vec{b}0^{N_n}\vec{b}\vec{b}0^{N_n-N_k}.$$

After that the UTM deletes the mark c (q_1c0Rq_4) and the second stage of modelling begins (first of all, the UTM writes the mark c in the part Q_{R}).

(ii) On the second stage of modelling:

$0\ L1$ $\quad(q_2\ 01\ \ Lq_2)$,
$\overleftarrow{b}\ L\vec{b}$ $\quad(q_2\ \overleftarrow{b}\vec{b}\ Lq_2)$,
$0\ bL\ 1c$ $\quad(q_2\ b\vec{b}\ Lq_3,\ \ q_3\ 0c\ Rq_1,\ q_1\ \vec{b}b\ Lq_1,\ q_1\ c0\ Rq_4,\ q_4\ bc\ Lq_2,\ q_2\ 01\ Lq_2)$.

$L\ \vec{b}(\overleftarrow{b})R$ $\ (q_2\ \vec{b}\overleftarrow{b}\ Rq_2)$ and the UTM writes symbol 1 in Q_{R}.

As after the first stage of modelling there are exactly N_r marks $\vec{b}$ between the codes P_r and S, the UTM will write exactly N_r symbols 1 in Q_r. After that, when the head is located on the code P_r, the UTM will write $N_n - N_r$ more symbols 1 in Q_r. As a result there will be the code A_n.

$L1(0)R$ $(q_2 10 Rq_2)$ and the UTM writes the symbol 1 in Q_R. In this case

$$\begin{array}{ll} 1\ R\ 0 & (q_2\ 10\ Rq_2), \\ \vec{b}\ R\ \overleftarrow{b} & (q_2\ \vec{b}\overleftarrow{b}\ Rq_2), \\ c\ R\ b & (q_2\ cb\ Rq_2), \\ R\ 0\ (1)L & (q_2\ 01\ Lq_2). \end{array}$$

$Lbb(\overleftarrow{b}\overleftarrow{b})R$ $(q_2\ b\ \vec{b}Lq_3,\ q_3\ b\ \overleftarrow{b}Rq_4,\ q_4\vec{b}\overleftarrow{b}Rq_4)$ and the UTM writes the mark c in Q_R. In this case

$$\begin{array}{ll} 1\ R\ 0 & (q_4\ 10\ Rq_4), \\ \vec{b}\ R\ \overleftarrow{b} & (q_4\ \vec{b}\overleftarrow{b}\ Rq_4), \\ c\ R\ b & (q_4\ cb\ Rq_4), \\ R\ 0\ (c)L & (q_4\ 0c\ Lq_2). \end{array}$$

If the head of the UTM moves to the left and meets the pair $\overleftarrow{b}b$, then the UTM halts $(q_2\ b\ \vec{b}Lq_3,\ q_3\overleftarrow{b}-)$.

If the head of the UTM moves to the left and meets the pair $1b$, the second stage of modelling is over. At that time the tape of the UTM is

$$Q_L P_{n+1} P_n \dots P_{r+1} P''_r P''_{r-1} \dots P''_1 P''_0 R'' A_t \dots A_w A_n A_n A_{r3} A_{r4} \dots A_{rm_r} Q_R$$

and the head is located on the second left symbol of the code P''_r. P''_k $(k \in \{0, \dots r\})$ differs from P_k by the fact that the marks b are changed to the marks $\vec{b}$.

Then $L1b(1b)R$ $(q_2\ b\vec{b}\ Lq_3,\ q_3\ 11\ Rq_3,\ q_3\ \vec{b}b\ Rq_3)$ and the UTM goes to the third stage of modelling.

(iii) On the third stage of modelling the UTM restores the tape in P:

$$\begin{array}{ll} \vec{b}\ R\ b & (q_3\ \vec{b}b\ Rq_3), \\ 1\ R\ 1 & (q_3\ 11\ Rq_3). \end{array}$$

When the head of the UTM moves to the right and meets the mark c, the third stage and the whole cycle of modelling are over. The UTM deletes the mark c and a new cycle of modelling $(q_3 c 1 Rq_1)$ begins.

9. The UTM with 3 states and 10 symbols

The symbols of the machine in UTM(3,10) (see [15]) are 0 (blank symbol), $1, \overleftarrow{1}, \vec{1}, b, \overleftarrow{b}, \vec{b}, c, \overleftarrow{c}, \vec{c}$; and the states are q_1, q_2, q_3.

$$N_1 = 1, \quad N_{k+1} = N_k + 2m_k + 2 \quad (k \in \{1, \dots, n\}).$$

The code of the production $\alpha_i = a_{i1}a_{i2}\dots a_{im_i}$ $(i \in \{1,\dots,n\})$ of the tag-system is

$$P_i = b1bbb1^{N_{im_i}}bb1^{N_{im_i-1}}\dots bb1^{N_{i2}}bb1^{N_{i1}},$$

where $A_j = 1^{N_j}$, $j \in \{1,\dots,n+1\}$ and the symbol b is a mark.

$$P_0 = b, \qquad P_{n+1} = \vec{c}b.$$

The code S of the initial word $\beta = a_r a_s a_t \dots a_w$, to be transformed by the tag-system, is

$$S = 1^{N_r}c1^{N_s}c1^{N_t}\dots c1^{N_w}c,$$

where the symbol c is a mark.

The program of the machine in UTM(3,10):

$q_1\ 0c\ Lq_3$	$q_2\ 0\vec{1}\ Lq_2$	$q_3\ 0\ $ —
$q_1\ b\overleftarrow{b}\ Rq_1$	$q_2\ b\vec{b}\ Lq_3$	$q_3\ b\overleftarrow{b}\ Rq_1$
$q_1\ \overleftarrow{b}b\ Lq_1$	$q_2\ \overleftarrow{b}\vec{b}\ Lq_2$	$q_3\ \overleftarrow{b}\vec{b}\ Lq_2$
$q_1\ \vec{b}\overleftarrow{b}\ Rq_1$	$q_2\ \vec{b}\overleftarrow{b}\ Rq_2$	$q_3\ \vec{b}b\ Rq_3$
$q_1\ 1\vec{1}\ Lq_1$	$q_2\ 1\overleftarrow{1}\ Rq_2$	$q_3\ 11\ Rq_3$
$q_1\ \vec{1}\overleftarrow{1}\ Rq_1$	$q_2\ \vec{1}\overleftarrow{1}\ Rq_2$	$q_3\ \vec{1}1\ Rq_3$
$q_1\ \overleftarrow{1}\vec{1}\ Lq_1$	$q_2\ \overleftarrow{1}\vec{1}\ Lq_2$	$q_3\ \overleftarrow{1}1\ Lq_3$
$q_1\ c\vec{1}\ Lq_2$	$q_2\ c\overleftarrow{c}\ Rq_2$	$q_3\ c1\ Rq_1$
$q_1\ \overleftarrow{c}\ $ —	$q_2\ \overleftarrow{c}\vec{c}\ Lq_2$	$q_3\ \overleftarrow{c}c\ Lq_3$
$q_1\ \vec{c}\overleftarrow{c}\ Rq_1$	$q_2\ \vec{c}\overleftarrow{c}\ Rq_2$	$q_3\ \vec{c}\ $ —

(i) On the first stage of modelling:

$1\ L\ \vec{1}$	$(q_1\ 1\vec{1}\ Lq_1)$,
$\overleftarrow{1}\ L\ \vec{1}$	$(q_1\ \overleftarrow{1}\vec{1}\ Lq_1)$,
$\overleftarrow{b}\ L\ b$	$(q_1\ \overleftarrow{b}b\ Lq_1)$,
$L\ b\ (\overleftarrow{b})R$	$(q_1\ b\overleftarrow{b}\ Rq_1)$,
$\vec{1}\ R\ \overleftarrow{1}$	$(q_1\ \vec{1}\overleftarrow{1}\ Rq_1)$,
$b\ R\ \overleftarrow{b}$	$(q_1\ b\overleftarrow{b}\ Rq_1)$,
$R\ 1\ (\vec{1})L$	$(q_1\ 1\vec{1}\ Lq_1)$.

If the head of the UTM moves to the right and meets the mark c, the first stage of modelling is over. The UTM deletes this mark and the second stage of modelling begins $(q_1\ c\ \vec{1}Lq_2)$.

(ii) On the second stage of modelling the UTM writes the marks c and the symbols $\vec{1}$ in Q_R; moreover the UTM writes the mark c only after the symbol $\vec{1}$.

If the UTM writes the symbol $\vec{1}$ in Q_R either after writing the mark c or after the first stage of modelling, then

$$
\begin{array}{ll}
\vec{b}\ R\ \overleftarrow{b} & (q_2\ \vec{b}\overleftarrow{b}\ Rq_2),\\
\vec{1}\ R\ \overleftarrow{1} & (q_2\ \vec{1}\overleftarrow{1}\ Rq_2),\\
1\ R\ \overleftarrow{1} & (q_2\ 1\overleftarrow{1}\ Rq_2),\\
\vec{c}\ R\ \overleftarrow{c} & (q_2\ \vec{c}\overleftarrow{c}\ Rq_2),\\
R\ 0\ (\vec{1})L & (q_2\ 0\vec{1}\ Lq_2).
\end{array}
$$

If the UTM writes the symbol $\vec{1}$ in Q_{R} after the same symbol $\vec{1}$, then

$$
\begin{array}{ll}
\vec{b}\ R\ \overleftarrow{b} & (q_2\ \vec{b}\overleftarrow{b}\ Rq_2),\\
\vec{1}\ R\ \overleftarrow{1} & (q_2\ \vec{1}\overleftarrow{1}\ Rq_2),\\
\vec{c}\ R\ \overleftarrow{c} & (q_2\ \vec{c}\overleftarrow{c}\ Rq_2),\\
R\ 0\ (\vec{1})L & (q_2\ 0\vec{1}\ Lq_2).
\end{array}
$$

The UTM goes to the left after writing the symbol $\vec{1}$ in Q_{R}:

$$
\begin{array}{ll}
\overleftarrow{1}\ L\ \vec{1} & (q_2\ \overleftarrow{1}\vec{1}\ Lq_2),\\
\overleftarrow{c}\ L\ \vec{c} & (q_2\ \overleftarrow{c}\vec{c}\ Lq_2),\\
\overleftarrow{b}\ L\ \vec{b} & (q_2\ \overleftarrow{b}\vec{b}\ Lq_2).
\end{array}
$$

Then, if the head of the UTM meets the symbol 1 in P_r, the head will change the direction of its motion, the UTM changes the symbol 1 to the symbol $\overleftarrow{1}$ and writes the symbol $\vec{1}$ in Q_{R}:

$$L\ 1(\overleftarrow{1})\ R\ \ (q_2\ 1\overleftarrow{1}\ Rq_2).$$

If the head of the UTM meets the marks bb in P_r, then the UTM writes the mark c in Q_{R}:

$$L\ bb\ (\overleftarrow{b}\overleftarrow{b})R\ (q_2\ b\ \vec{b}Lq_3,\ \ q_3\ b\ \overleftarrow{b}Rq_1,\ \ q_1\vec{b}\,\overleftarrow{b}Rq_1),\ \text{then}$$

$$
\begin{array}{ll}
\vec{b}\ R\ \overleftarrow{b} & (q_1\ \vec{b}\overleftarrow{b}\ Rq_1),\\
\vec{1}\ R\ \overleftarrow{1} & (q_1\ \vec{1}\overleftarrow{1}\ Rq_1),\\
\vec{c}\ R\ \overleftarrow{c} & (q_1\ \vec{c}\overleftarrow{c}\ Rq_1),\\
R\ 0\ (c)L & (q_1\ 0c\ \ Lq_3).
\end{array}
$$

When the head of the UTM moves to the left after writing the symbol c in Q_{R}, then

$$
\begin{array}{ll}
\overleftarrow{1}\ L\ 1 & (q_3\ \overleftarrow{1}1\ Lq_3),\\
\overleftarrow{c}\ L\ c & (q_3\ \overleftarrow{c}c\ Lq_3),
\end{array}
$$

and the UTM restores the part S of the tape. Then the UTM meets the mark $\overleftarrow{b}$ in P:

$$\overleftarrow{b}\ L\ \vec{b} \quad (q_3\ \overleftarrow{b}\vec{b}\ Lq_2,\ \ q_2\ \overleftarrow{b}\vec{b}\ Lq_2),$$
$$\overleftarrow{1}\ L\ \vec{1} \quad (q_2\ \overleftarrow{1}\vec{1}\ Lq_2).$$

The UTM halts when it meets the pair $\vec{c}\,b$ $(q_2\ b\overrightarrow{L}\ q_3,\ \ q_3\vec{c}-)$.

If the UTM meets the pair $1b$, then the second stage of modelling is over. Then

$$L\ 1b(1b) \quad (q_2\ b\vec{b}\ Lq_3,\ q_3\ 11\ Rq_3,\ q_3\ \vec{b}b\ Rq_3)$$

and the third stage of modelling begins.

(iii) On the third stage of modelling the UTM restores the tape in P (the tape is restored in S after writing the mark c in Q_{R}):

$$\vec{1}\ R\ 1 \quad (q_3\ \vec{1}1\ Rq_3),$$
$$1\ R\ 1 \quad (q_3\ 11\ Rq_3),$$
$$\vec{b}\ R\ b \quad (q_3\ \vec{b}b\ Rq_3).$$

When the head of the UTM moves to the right and meets the mark c, both the third stage and the whole cycle of modelling are over. The UTM deletes the mark c and a new cycle of modelling begins (q_3c1Rq_1).

10. The UTM with 2 states and 18 symbols

The symbols of the machine in UTM(2,18) are $\overleftarrow{1}$ (blank symbol), $1, \vec{1}, \overleftarrow{1}_1, \vec{1}_1, b, \overleftarrow{b}, \vec{b}, \overleftarrow{b}_1, \vec{b}_1, b_2, b_3, c, \overleftarrow{c}, \vec{c}, \overleftarrow{c}_1, \vec{c}_1$ and c_2; and the states are q_1 and q_2.

$$N_1 = 1, \quad N_{k+1} = N_k + m_k + 1 \quad (k \in \{1,\ldots,n\}).$$

The code of the production $\alpha_i = a_{i1}a_{i2}\ldots a_{im_i}$ $(i \in \{1,\ldots,n\})$ of the tag-system is

$$P_i = bb1^{N_{im_i}}1b1^{N_{im_i-1}}\ldots 1b1^{N_{i2}}1b1^{N_{i1}},$$

where $A_j = 1^{N_j}$, $j \in \{1,\ldots,n+1\}$ and the symbol b is a mark.

$$P_0 = bb, \quad P_{n+1} = \overleftarrow{c}_1\overleftarrow{c}_1.$$

The code S of the initial word $\beta = a_ra_sa_t\ldots a_w$, to be transformed by the tag-system, is

$$S = 1^{N_r}c1^{N_s}c1^{N_t}\ldots c1^{N_w}c,$$

and the symbol c is a mark.

The program of the machine in UTM(2,18):

q_1	$1c_2$	Lq_1	q_2	$1\overleftarrow{1}$	Rq_2
q_1	$\vec{1}\overleftarrow{1}_1$	Rq_1	q_2	$\vec{1}\overleftarrow{1}$	Rq_2
q_1	$\overleftarrow{1}c_2$	Lq_1	q_2	$\overleftarrow{1}\vec{1}$	Lq_2
q_1	$\vec{1}_1 1$	Rq_1	q_2	$\vec{1}_1\overleftarrow{1}_1$	Rq_2
q_1	$\overleftarrow{1}_1\vec{1}_1$	Lq_1	q_2	$\overleftarrow{1}_1 1$	Lq_2
q_1	$b\overleftarrow{b}$	Rq_1	q_2	bb_2	Rq_1
q_1	$\vec{b}\overleftarrow{b}_1$	Rq_1	q_2	$\vec{b}\overleftarrow{b}$	Rq_2
q_1	$\overleftarrow{b}b$	Lq_1	q_2	$\overleftarrow{b}\vec{b}$	Lq_2
q_1	$\vec{b}_1 b$	Rq_1	q_2	$\vec{b}_1\overleftarrow{b}_1$	Rq_2
q_1	$\overleftarrow{b}_1\vec{b}_1$	Lq_1	q_2	$\overleftarrow{b}_1\vec{b}$	Lq_2
q_1	b_2b_3	Lq_2	q_2	b_2b	Rq_1
q_1	$b_3\vec{b}_1$	Lq_2	q_2	$b_3\overleftarrow{b}_1$	Rq_2
q_1	$c\vec{1}$	Lq_2	q_2	$c\overleftarrow{c}$	Rq_2
q_1	$\vec{c}\overleftarrow{c}$	Rq_1	q_2	$\vec{c}\overleftarrow{c}$	Rq_2
q_1	$\overleftarrow{c}\vec{c}_1$	Lq_1	q_2	$\overleftarrow{c}\vec{c}$	Lq_2
q_1	$\vec{c}_1\overleftarrow{c}_1$	Rq_2	q_2	$\vec{c}_1 c_2$	Rq_2
q_1	$\overleftarrow{c}_1$	—	q_2	$\overleftarrow{c}_1 c_2$	Lq_1
q_1	$c_2\overleftarrow{1}$	Rq_1	q_2	c_2c	Lq_2

(i) On the first stage of modelling:

$1 \ L \ c_2$	$(q_1 \ 1c_2 \ Lq_1)$,
$\overleftarrow{1} \ L \ c_2$	$(q_1 \ \overleftarrow{1}c_2 \ Lq_1)$,
$\overleftarrow{b} \ L \ b$	$(q_1 \ \overleftarrow{b}b \ Lq_1)$,
$L \ b \ (\overleftarrow{b})R$	$(q_1 \ b\overleftarrow{b} \ Rq_1)$,
$c_2 \ R \ \overleftarrow{1}$	$(q_1 \ c_2\overleftarrow{1} \ Rq_1)$,
$b \ R \ \overleftarrow{b}$	$(q_1 \ b\overleftarrow{b} \ Rq_1)$,
$R \ 1 \ (c_2)L$	$(q_1 \ 1c_2 \ Lq_1)$.

If the head of the UTM moves to the right and meets the mark c, then the first stage of modelling is over. The UTM deletes this mark and the second stage of modelling begins ($q_1 \ c \ \vec{1}Lq_2$). The tape of the UTM is

$$Q_{\mathrm{L}}P_{n+1}P_n \ldots P_{r+1}P_rP'_{r-1} \ldots P_1{}'P_0{}'R'A_t \ldots A_wQ_{\mathrm{R}},$$

where in $P_i{}'$ ($i \in \{0,\ldots,r-1\}$) the 1 symbols are replaced by $\overleftarrow{1}$ and the b marks are replaced by $\overleftarrow{b}$, R' consists of $\overleftarrow{1}$ and $\vec{1}$ and the head of the UTM is located on the R' in the state q_2.

(ii) On the second stage of modelling the UTM writes the marks c_2 and the symbols $\vec{1}$ in Q_{R}; moreover, the UTM writes the mark c_2 only after the symbol $\vec{1}$.

If the head of the UTM moves to the left in the code P_r and meets the symbol 1, then the head will change the direction of its motion, the UTM changes the symbol 1 to the symbol $\overleftarrow{1}$ and writes the symbol $\overrightarrow{1}$ in Q_R:

$$L\ 1(\overleftarrow{1})\ R \quad (q_2\ 1\overleftarrow{1}\ Rq_2).$$

If the UTM writes the symbol $\overrightarrow{1}$ in Q_R either after writing the mark c_2 or after the first stage of modelling, then

$$\begin{array}{ll}
\overrightarrow{b}\ R\ \overleftarrow{b} & (q_2\ \overrightarrow{b}\overleftarrow{b}\ Rq_2), \\
\overrightarrow{1}\ R\ \overleftarrow{1} & (q_2\ \overrightarrow{1}\overleftarrow{1}\ Rq_2), \\
1\ R\ \overleftarrow{1} & (q_2\ 1\overleftarrow{1}\ Rq_2), \\
c\ R\ \overleftarrow{c} & (q_2\ c\overleftarrow{c}\ Rq_2), \\
R\ \overleftarrow{1}\ (\overrightarrow{1})L & (q_2\ \overleftarrow{1}\overrightarrow{1}\ Lq_2).
\end{array}$$

If the UTM writes the symbol $\overrightarrow{1}$ in Q_R after the same symbol $\overrightarrow{1}$, then

$$\begin{array}{ll}
\overrightarrow{b}\ R\ \overleftarrow{b} & (q_2\ \overrightarrow{b}\overleftarrow{b}\ Rq_2), \\
\overrightarrow{1}\ R\ \overleftarrow{1} & (q_2\ \overrightarrow{1}\overleftarrow{1}\ Rq_2), \\
\overrightarrow{c}\ R\ \overleftarrow{c} & (q_2\ \overrightarrow{c}\overleftarrow{c}\ Rq_2), \\
R\ \overleftarrow{1}\ (\overrightarrow{1})L & (q_2\ \overleftarrow{1}\overrightarrow{1}\ Lq_2).
\end{array}$$

When the head of the UTM moves to the left having written the symbol $\overrightarrow{1}$ in Q_R, then

$$\begin{array}{ll}
\overleftarrow{1}\ L\ \overrightarrow{1} & (q_2\ \overleftarrow{1}\overrightarrow{1}\ Lq_2), \\
\overleftarrow{c}\ L\ \overrightarrow{c} & (q_2\ \overleftarrow{c}\overrightarrow{c}\ Lq_2), \\
\overleftarrow{b}\ L\ \overrightarrow{b} & (q_2\ \overleftarrow{b}\overrightarrow{b}\ Lq_2).
\end{array}$$

If the head of the UTM moves to the left in the code P_r after writing the symbol $\overrightarrow{1}$ in Q_R and meets the mark b, then the UTM writes the mark c_2 in Q_R:

$$\begin{array}{ll}
L\ b\ (b_2)R & (q_2\ bb_2\ Rq_1), \\
\overrightarrow{b}\ R\ \overleftarrow{b}_1 & (q_1\ \overrightarrow{b}\overleftarrow{b}_1\ Rq_1), \\
\overrightarrow{1}\ R\ \overleftarrow{1}_1 & (q_1\ \overrightarrow{1}\overleftarrow{1}_1\ Rq_1), \\
\overrightarrow{c}\ R\ \overleftarrow{c} & (q_1\ \overrightarrow{c}\overleftarrow{c}\ Rq_1), \\
R\ \overleftarrow{1}\ (c_2)L & (q_1\ \overleftarrow{1}c_2\ Lq_1).
\end{array}$$

When the head of the UTM moves to the left after writing the symbol c_2 in Q_R, then

$$\begin{array}{ll}
\overleftarrow{1}_1\ L\ \overrightarrow{1}_1 & (q_1\ \overleftarrow{1}_1\overrightarrow{1}_1\ Lq_1), \\
\overleftarrow{c}\ L\ \overrightarrow{c}_1 & (q_1\ \overleftarrow{c}\overrightarrow{c}_1\ Lq_1), \\
\overleftarrow{b}_1\ L\ \overrightarrow{b}_1 & (q_1\ \overleftarrow{b}_1\overrightarrow{b}_1\ Lq_1).
\end{array}$$

The UTM halts on the second stage when it locates the symbol $\overleftarrow{c}_1$ in the state q_1:

$$(q_2\ \overleftarrow{c}_1c_2\ Lq_1,\ \ q_1\ \overleftarrow{c}_1-).$$

Now there are two variants when the head of the UTM moves to the left and meets the pairs $1b_2$ or bb_2:

(a) $L1b_2(\overleftarrow{1}\overleftarrow{b}_1)R$ $(q_1b_2b_3Lq_2,\ q_2\ 1\ \overleftarrow{1}Rq_2,\ q_2\ b_3\ \overleftarrow{b}_1Rq_2)$ and the UTM restores the part S of the tape and the second stage continues:

$$\begin{array}{llll}
\vec{1}_1\ R\ \overleftarrow{1}_1 & (q_2\ \vec{1}_1\overleftarrow{1}_1\ Rq_2), \\
\vec{b}_1\ R\ \overleftarrow{b}_1 & (q_2\ \vec{b}_1\overleftarrow{b}_1\ Rq_2), \\
\vec{c}_1\ R\ c_2 & (q_2\ \vec{c}_1c_2\ Rq_2), \\
R\ c_2\ (c)L & (q_2\ c_2c\ Lq_2), \\
\overleftarrow{1}_1\ L\ 1 & (q_2\ \overleftarrow{1}_11\ Lq_2), \\
c_2\ L\ c & (q_2\ c_2c\ Lq_2), \\
\overleftarrow{b}_1\ L\ \vec{b} & (q_2\ \overleftarrow{b}_1\vec{b}\ Lq_2).
\end{array}$$

Now the UTM can write the symbol $\vec{1}$ in Q_{R}.

(b) $Lbb_2(bb)R$ $(q_1b_2b_3Lq_2,\ q_2bb_2Rq_1,\ q_1\ b_3\ \overleftarrow{b}_1Lq_2,\ q_2b_2bRq_1,\ q_1\vec{b}_1bRq_1)$ and the third stage begins:

(iii) On the third stage of modelling the UTM restores the tape in P:

$$\begin{array}{ll}
\vec{1}_1\ R\ 1 & (q_1\ \vec{1}_11\ Rq_1), \\
\vec{b}_1\ R\ b & (q_1\ \vec{b}_1b\ Rq_1).
\end{array}$$

When the head of the UTM moves to the right and meets the mark $\vec{c}_1$, then this mark is changed to $\overleftarrow{c}_1$ $(q_1\vec{c}_1\overleftarrow{c}_1Rq_2)$ and, then, the UTM restores the tape in S:

$$\begin{array}{ll}
\vec{1}_1\ R\ \overleftarrow{1}_1 & (q_2\ \vec{1}_1\overleftarrow{1}_1\ Rq_2), \\
\vec{c}_1\ R\ c_2 & (q_2\ \vec{c}_1c_2\ Rq_2), \\
R\ c_2\ (c)L & (q_2\ c_2c\ Lq_2), \\
\overleftarrow{1}_1\ L\ 1 & (q_2\ \overleftarrow{1}_11\ Lq_2), \\
c_2\ L\ c & (q_2\ c_2c\ Lq_2).
\end{array}$$

When the head of the UTM moves to the left and meets the mark $\overleftarrow{c}_1$, both the third stage and the whole cycle of modelling are over. The UTM deletes the mark $\overleftarrow{c}_1$ and a new cycle of modelling begins $(q_2\ \ \overleftarrow{c}_1c_2\ \ Lq_1)$.

Acknowledgements

The author is grateful to Dr. Alexei Yu. Muravitsky (Courant Institute of Mathematical Sciences, NY) and referees for help, useful discussions and comments.

References

[1] M. Davis, The definition of the universal Turing machine, *Proc. Amer. Math. Soc.* **8** (1957) 1125–1126.

[2] M.D. Davis and E.J. Weyuker, *Computability, Complexity, and Languages* (Academic Press, New York, 1983).

[3] V. Diekert and M. Kudlek, Small deterministic Turing machines, Papers on Automata and Languages, Dep. of.Math., Karl Marx Univ. of Economics, Budapest, 188-4, 1989, 77–87.
[4] G.T. Herman, A new hierarchy of elementary functions, *Proc. Amer. Math. Soc.* **20** (1969) 557–562.
[5] M. Kudlek, Small deterministic Turing machines, *Theoret. Comput. Sci.* **168** (this Vol.) (1996) 241–255.
[6] A.I. Maltsev, *The Algorithms and Recursive Functions* (Moskva, Nauka, 2nd ed., 1986) (Russian).
[7] M. Margenstern, Non erasing Turing machines: a frontier between a decidable halting problem and universality, in: *Proc. FCT'93*, Lecture Notes in Computer Science, Vol. 710 (1993) 375–385.
[8] M.L. Minsky, Size and structure of universal Turing machines using tag systems, in: *Recursive Function Theory, Symp. in Pure Mathematics*, Vol. 5 (Amer. Mathematical Soc., Provelence, RS, 1962) 229–238.
[9] A. Nozaki, On notion of universality of turing machine, *Kybernetika* **5** (1969) 29–42.
[10] L. M. Pavlotskaya, *Problemi kibernetiki*, Vol. 33 (Moskva, Nauka, 1978) 91–118 (Russian).
[11] L. Priese, Towards a precise characterization of the complexity of the universal and nonuniversal Turing machines, *SIAM J. Comput.* **8** (1979) 508–523.
[12] R.M. Robinson, Minsky's small universal Turing machine, *Internat. J. Math.* **2** (1991) 551–562.
[13] Yu.V. Rogozhin, Seven universal Turing machines, in: *Systems and Theoretical Programming*, Mat. Issled. no.69, Academiya Nauk Moldavskoi SSR, (Kishinev, 1982) 76–90 (Russian).
[14] Yu.V. Rogozhin, A universal Turing machine with 10 states and 3 symbols, *Izvestiya Akademii Nauk Respubliki Moldova, Matematika* **4** (10) (1992) 80–82 (Russian).
[15] Yu.V. Rogozhin, About Shannon's problem for Turing machines, *Comput. Sci. J. Moldova* **1** 3(3) (1993) 108–111.
[16] Yu.V. Rogozhin, On the notion of universality and Shannon's problem for Turing machines, MCU/UMC'95, *International Workshop 'Universal Machines and Computations'*, Lecture Abstracts (Paris, 29–31 March 1995), 36–39.
[17] C.E. Shannon, A universal Turing machine with two internal states, in: *Automata Studies*, Ann. Math. Stud., Vol. 34, (Princeton Univ.Press, 1956) Princeton, 157–165.