The Equivalence Problem of Deterministic Multitape Finite Automata: A New Proof of Solvability Using a Multidimensional Tape

Alexander A. Letichevsky[1], Arsen S. Shoukourian[2], and Samvel K. Shoukourian[3]

[1] National Academy of Sciences of Ukraine, Glushkov Institute of Cybernetics
let@cyfra.net
[2] National Academy of Sciences of Armenia, Institute for Informatics and Automation Problems
arsen.shoukourian@gmail.com
[3] Yerevan State University, IT Educational and Research Center
samshouk@sci.am

Abstract. This publication presents a new proof of solvability for the equivalence problem of deterministic multitape finite automata, based on modeling their behavior via a multidimensional tape. It is shown that for a decision on equivalence of two automata it is necessary and sufficient to consider finite sets of their execution trace words built over the mentioned multidimensional tape.

1 Introduction

Deterministic multitape finite automata and their equivalence problem were introduced by M. O. Rabin and D. Scott in 1959 [6]. The solvability of the equivalence problem for two tape automata was proven by M. Bird in 1973 [1]. The way of solution is based on equivalent transformation of source automata graphs to a special finite commutative diagram, which takes into account commutativity assumptions for symbols of the alphabet used in different tapes. Meantime, the suggested way is not acceptable for the case when the number of tapes is more than two, because, for that case, the suggested transformation may not lead to a finite commutative diagram. In 1993 T. Harju and J. Karhumäki proved the solvability of the equivalence problem for multitape automata without any restriction on the number of tapes [5]. It is based on a purely algebraic technique. Per T. Harju "the main argument of the proof uses an embedding result for ordered groups into division rings".

A new combinatorial proof of solvability is presented in this article. It is similar to the solution suggested by M. Bird, but instead of a transformation of source automata to a commutative diagram, the commutativity assumptions are taken into account via a special multidimensional tape [3][4] used for coding execution

A.-H. Dediu, H. Fernau, and C. Martín-Vide (Eds.): LATA 2010, LNCS 6031, pp. 392–402, 2010.

traces of source automata. The advantages and disadvantages of the proposed solution imply from its combinatorial nature.

A special representation of an element in a partially commutative semigroup is described in section 2. The new proof of solvability of the equivalence problem of multitape automata is adduced in section 3.

This is the third new application of multidimensional tapes for solving problems in theory of automata. The first applications were in [4][7] for problems open for many years.

2 Partially Commutative Semigroups

If X is an alphabet, then the semigroup of all words in the alphabet X, including the empty word, will be denoted by F_X, and the semigroup of all n-element vectors of words will be denoted by F_X^n.

Let G be a semigroup with a unit, generated by the set of generators $Y = \{y_1, \dots, y_n\}$. G is called *free partially commutative semigroup*, if it is defined by a finite set of definitive assumptions of type $y_i y_j = y_j y_i$ [2].

Let $K : F_Y \longrightarrow F^n_{\{0,1\}}$ be a homomorphism over the semigroup F_Y which maps words from F_Y to n-element vectors in binary alphabet $\{0, 1\}$ [3]. The homomorphism K over the set of generators of the semigroup F_Y is defined by the equation

$$K(y_i) = (\alpha_{1i}, \dots, \alpha_{ni}), \quad \text{where} \quad \alpha_{ij} = \begin{cases} 1, \, i = j \\ e, \, y_i y_j = y_j y_i \\ 0, \, y_i y_j \neq y_j y_i \end{cases} \tag{1}$$

At the same time $K(e) = (e, \dots, e)$.

Lemma 1. *Let y_i, y_j be generators of G, $y_i \neq y_j$, $g_1 = y_i y_j$, $g_2 = y_j y_i$ be elements of G, obtained after applying the operation of the semigroup G to generators y_i and y_j. $g_1 = g_2 \Longleftrightarrow K(y_i y_j) = K(y_j y_i)$* [3].

This statement allows to consider the homomorphism K as a mapping not only over the semigroup of words F_Y, but also over the free partially commutative semigroup G.

A *linear order* $<$ is specified over the set of generators Y [3]. This order *coincides with the order of enumeration of generators* (lexicographical order). The order $<$ is used for sorting the representations of elements of the semigroup G and for defining their canonical representation.

Let $h \in F_Y$. A word in the alphabet $\{0, 1\}$ is called a mask of occurrences of a generator y in the word h, if it is ensued from the substitution of all occurrences of the generator y by 1, and from the substitution of occurrences of all other generators by 0. If the masks are considered not only as words in the alphabet $\{0, 1\}$, but also as binary integers, then it will be possible to compare them. It is assumed that the lexicographical order defines the order for comparing binary representations of generators.

Let g be an element of the semigroup G, and $h, q \in F_Y$ be its representations. The representation h is less than the representation q, $h < q$, if there exists such a generator y that:

1. if $y' < y$ then masks of occurrences of the generator y' in words h and q are equal,
2. the mask of occurrences of the generator y in the word h is less than the mask of occurrences of that generator in the word q.

Lemma 2. *Any two different representations of the same element of the semigroup G are comparable in terms of the order* $<$ [3].

It is implied, from Lemma 2, that there exists a minimal representation for every element of G. That representation is called the *canonical form* of the element, induced by the order $<$ over the set of generators [3]. The following property of the canonical representation of an element is true: any two symbols that stand next to each other in the canonical representation are either not commutative or the left one is less (in terms of the order $<$) than the right one.

An equivalence relation ρ over the semigroup F_Y is specified as follows. If w_1 and w_2 are words from F_Y, $w_1, w_2 \in F_Y$, then $w_1 \rho w_2$ if and only if w_1 coincides with w_2 up to the commutativity assumptions.

The relation ρ partitions the semigroup F_Y to disjoint classes. These classes will further be called *classes of commutation.* It is obvious, that a commutation class C_g corresponds to an element g from the semigroup G. An element from C_g, which is the canonical form of g, in turn, will be called the *representative of the commutation class* C_g.

Lemma 3. *Any free partially commutative semigroup of n generators is isomorphic to some sub-semigroup of Cartesian product of n free semigroups with two generators* [3].

Let $Y = \{y_1, y_2\}$, $y_1 y_2 \neq y_2 y_1$. Using the binary coding considered above, i.e. $(1, 0)$ for y_1 and $(0, 1)$ for y_2, the following correspondence between words in F_Y

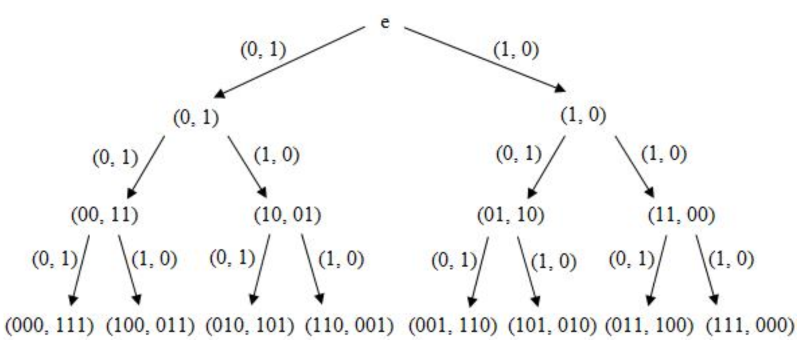

Fig. 1. Correspondence between the words in F_Y and vectors in binary alphabet $\{0, 1\}$

and two-element vectors in binary alphabet $\{0, 1\}$ (see Fig. 1) will be obtained: $K(e) = (e, e)$, $K(y_1) = (1, 0)$, $K(y_2) = (0, 1)$, $K(y_1y_1) = (11, 00) = (3, 0)$, $K(y_1y_2) = (01, 10) = (1, 2)$, $K(y_2y_1) = (10, 01) = (2, 1)$, $K(y_2y_2) = (00, 11) = (0, 3)$, ...

As it was mentioned above each element of a vector can be considered as an integer. Thus, a new correspondence between words in F_Y and elements of N^2, where $N = \{0, 1, \ldots\}$ and (e, e) corresponds to $(0, 0)$, is obtained (see Fig. 2). The elements of N^2 which correspond to words in F_Y are located on diagonals marked bold in the Fig. 2.

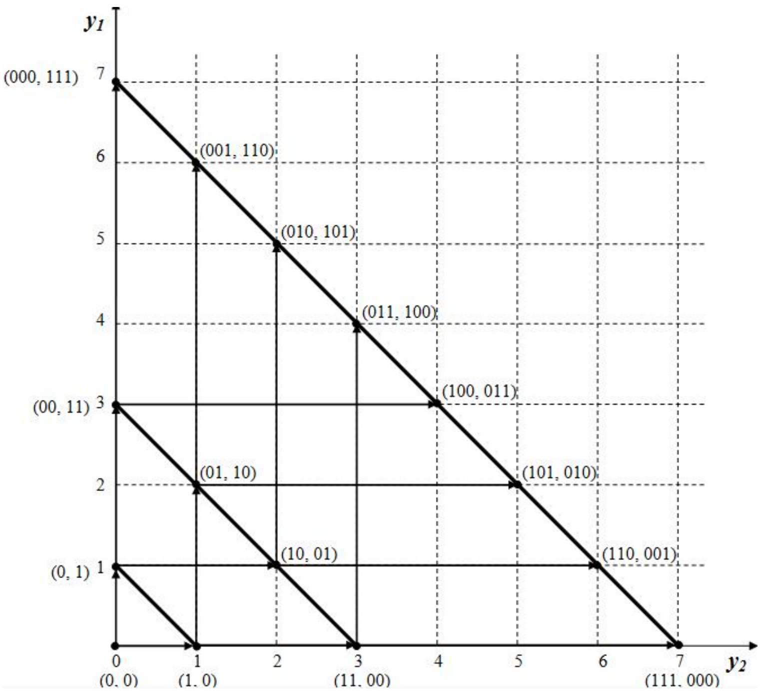

Fig. 2. Correspondence between the words in F_Y and elements of N^2

The correspondence between words in F_Y and elements of N^3, in more complicated case when $Y = \{y_1, y_2, y_3\}$, $y_1y_3 = y_3y_1$, $y_2y_3 = y_3y_2$, $y_1y_2 \neq y_2y_1$ is depicted in the Fig. 3.

Thus, introduction of the described binary coding for elements of a partially commutative semigroup and its justification in Lemmas 1, 2 and 3 allows to consider cells of a multidimensional tape introduced below instead of semigroup elements and, therefore, to avail of the opportunity to compare them as integer vectors when analyzing behaviors of two automata on a given element of the semigroup. Specifically, it becomes possible to measure the distance between

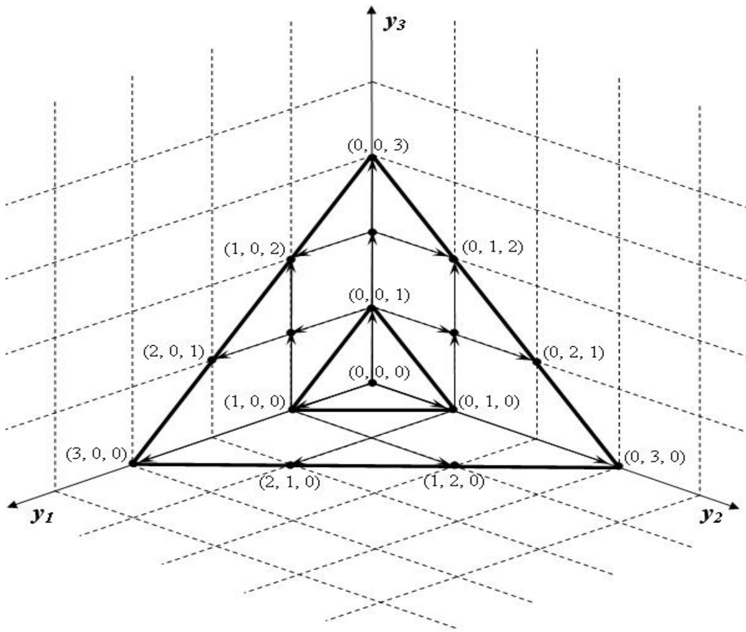

Fig. 3. Correspondence between the words in F_Y and elements of N^3

two automata during their movement on different representations of the same element of the semigroup.

3 A New Proof of Solvability

Some definitions from [3][4], which are necessary for further consideration, will be repeated below.

Let r be a positive integer, $N = \{0, 1, \ldots\}$. The set N^r is called an *r-dimensional tape*. Any element of N^r - $(a_1, \ldots, a_r)$ is called a *cell* of the tape and the numbers $a_1, \ldots, a_r$ are called the *coordinates* of the corresponding cell. The cell $(0, \ldots, 0)$ is the *initial* cell. Let Y be a finite alphabet. Any mapping $N^r \longrightarrow Y$ is called a fill of the tape with the symbols of Y.

It is considered that the alphabet Y is ordered $Y = \{y_1, \ldots, y_n\}$. It is divided into disjoint ordered subsets $Y = \bigcup\limits_{i=1}^{p} Y_i$, $Y_i \bigcap\limits_{i \neq j} Y_j = \emptyset$, preserving the given order for Y: $Y_i = \{y_{f(i)}, \ldots, y_{f(i)+|Y_i|-1}\}$ where $|Y_i|$ is the number of elements in Y_i, $f(1) = 1$, $f(i+1) = f(i) + |Y_i|$ when $i > 1$.

The n-dimensional tape N^n is considered, $n = |Y_1| + \ldots + |Y_p|$. Each $|Y_i|$ dimensions are used for expressing the movement on symbols from Y_i. The subset of dimensions corresponding to Y_i will be denoted by $D(Y_i)$. The position of the first coordinate corresponding to the subset $D(Y_i)$ is denoted by $f(i)$.

Correspondingly, the position of the last coordinate for the subset of dimensions $D(Y_i)$ can be expressed as $f(i) + |Y_i| - 1$. Several definitions are adduced below.

Suppose that $A =< Y, S, \delta, F, s_0 >$ is a deterministic p-tape automaton with the input alphabet Y, has a set $S = S_1 \bigcup \ldots \bigcup S_p$ as the set of states, $S_i \bigcap_{i \neq j} S_j = \emptyset$, δ as the completely defined transition function, F as the set of final states, and s_0 as the initial state. Y_i is the alphabet of the tape i, S_i is the set of states for the head i.

The notion of a predecessor is naturally demonstrated via binary representations with variable length for coordinates of cells adduced in the section 2. The binary representation of the initial cell consists of one digit codes: $(0, \ldots, 0)$. The binary representation of any other cell is built basing on the binary representation of a predecessor for the considered cell, because each successor has only one predecessor here. The length of any coordinate code for the successor is one more than the length of that coordinate code for the predecessor. At the same time, for a given coordinate, only one successor among successors of a given predecessor has the most left bit of the binary representation equal to 1.

A cell $a_1 = (\alpha_{11}, \ldots, \alpha_{1n})$ is called a *predecessor* of a cell $a_2 = (\alpha_{21}, \ldots, \alpha_{2n})$ if and only if there exists a number $j \in \{1, \ldots, p\}$, such that for any k from $\{1, \ldots, f(j) - 1, f(j) + |Y_j|, f(j) + |Y_j| + 1, \ldots, n\}$, $\alpha_{2k} = \alpha_{1k}$ and $\exists l \in \{f(j), \ldots, f(j) + |Y_j| - 1\}$ that $\forall m \in \{f(j), \ldots, f(j) + |Y_j| - 1\}$

1. $\alpha_{2m} = \alpha_{1m}, m \neq l,$
2. $\alpha_{2m} = \alpha_{1m} + (L + 1),\ m = l,$ where $L = \alpha_{11} + \ldots + \alpha_{1n}$.

Two predicates are introduced to represent if one cell is a predecessor of another. $\pi(a_1, a_2)$ is true if and only if the cell a_1 is the predecessor of the cell a_2. Another predicate $\pi_q(a_1, a_2)$ is introduced also. It is true if and only if the predicate $\pi(a_1, a_2)$ is true and a_1, a_2 differ only by the value of the coordinate q as follows $\alpha_{2q} = \alpha_{1q} + (L + 1)$, where $L = \alpha_{11} + \ldots + \alpha_{1n}$.

For a given automaton A a partial mapping $\phi_A : N^n \longrightarrow S$ is introduced:

1. $\phi_A(0, \ldots, 0) = s_0$
2. $\forall a \in N^n \setminus (0, \ldots, 0)\ \exists j \in \{1, \ldots, n\}\ \exists a^{(j)}_{pred}$ that $\pi_j(a^{(j)}_{pred}, a)$ is true and $\phi_A(a^{(j)}_{pred})$ is defined $\Longrightarrow \phi_A(a) = \delta(\phi_A(a^{(j)}_{pred}), y_j)$.
3. $\phi_A(a)$ is considered defined if and only if it is defined according to points 1 and 2 above.

Lemma 4. *For a given automaton A and a given cell $a \in N^n \setminus (0, \ldots, 0)$ if $\phi_A(a)$ is defined then there exists a unique cell a' such that $\phi_A(a')$ is defined and $\pi(a', a)$ is true.*

The graph of the mapping ϕ_A will be named a *set of all execution traces* for the automaton A.

It is evident that there exists such a mapping for any automaton with completely defined transition function.

A part of a set of execution traces, where the sum of coordinates of each cell is less or equal to $k - 1$ is called a *trace word* of the length k. The set of all trace words will be denoted further by Ω_A. The set of all cells used in a given trace word ω will be denoted by U_ω.

The part of a trace word ω, where the sum of coordinates of each cell is equal to k, is called the k^{th} *diagonal* of the word ω and is denoted by $d_k(\omega)$. The set of all cells used in a given diagonal d will be denoted by U_d. The length of $d_k(\omega)$ is equal to k+1.

The length of a trace word ω is equal to the number of diagonals it contains and is denoted by $length(\omega)$.

The diagonals which lengths are less by one than the powers of 2 : $2^0 - 1$, $2^1 - 1$, ... will be named *essential*.

The set of all essential diagonals is denoted by $D^{(E)} = \{d_h^{(E)} | h = 0, 1, \ldots\}$, and the length of a given essential diagonal $l_{d_{h+1}^{(E)}} = l_{d_h^{(E)}} + 2^h, l_{d_0^{(E)}} = 0$.

A trace word for a 3-tape automaton is adduced in Figure 4.

For a given trace word ω a path $p = a_{p_1} \ldots a_{p_m}$, $m \geq 1$, $a_{p_j} \in U_\omega$, $j \in \{1, \ldots, m\}$, is defined as a sequence of cells for which $\pi(a_{p_v}, a_{p_{v+1}})$ is true, $v = 1, \ldots, m-1$.

For a given path $p = a_{p_1} \ldots a_{p_m}$ a word $\chi_p = y_{q_1} \ldots y_{q_m - 1}$ in the alphabet Y is called a *characteristics* of the path p if and only if $\forall j \in \{1, \ldots, m\}$, $\pi_{q_j}(a_{p_j}, a_{p_{j+1}})$ is true.

A path $p = a_{p_1} \ldots a_{p_m}$ is called *complete* if $a_{p_1} = (0, \ldots, 0)$. A complete path $p = a_{p_1} \ldots a_{p_m}$ will be called accepted by a given automaton A if $\phi_A(a_{p_m})$ is a final state of A. Basing on considerations in section 2 we can consider the binary

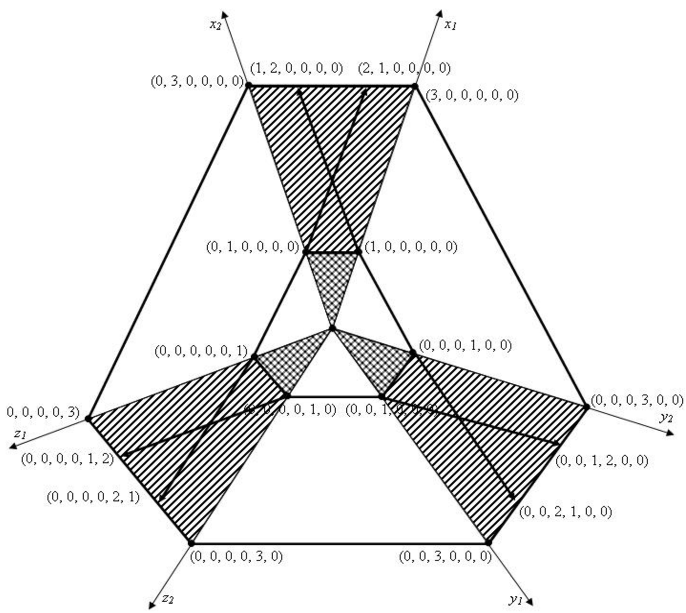

Fig. 4. A trace word for a 3-tape automaton over a two-symbol alphabet for each tape

coding of the characteristics $\chi_p \in F_Y$ for the path p. Its canonical representation coincides with the binary representation of a_{p_m} coordinates. Thus, the binary representation of a_{p_m} coordinates can be considered as a *canonical form* of the complete path $p = a_{p_1} \dots a_{p_m}$.

The set of all accepted paths in a trace word ω for a given automaton A is denoted by $AP_A(\omega)$ and the set of their canonical forms is denoted by $CF_{AP_A(\omega)}$.

Suppose A_1 and A_2 are multitape automata. Suppose also that $k = s^2 \times (2^s - 1)$, $s = |S_1| + |S_2|$, where S_j is the set of states of the automaton A_j, $j = 1, 2$, and ω_j is a trace word, $length(\omega_j) = k$.

Let A be an automaton. It is evident that $\forall d_h^{(E)} \in D^{(E)}\ \exists a \in U_{d_h^{(E)}}$ that $\phi_A(a)$ is defined.

A new mapping $\psi_A : D^{(E)} \longrightarrow 2^{N^n \times S}$ is introduced in the following way: $\psi_A(d_h^{(E)}) = \bigcup\limits_{a \in U_{d_h^{(E)}}} (a, \phi_A(a)),\ \forall d_h^{(E)} \in D^{(E)}$.

A subset of $\psi_A(d_h^{(E)})$ containing all pairs with final states and only them is denoted by $\xi_A(d_h^{(E)})$.

Set of all first components of $\psi_A(d_h^{(E)})$ and $\xi_A(d_h^{(E)})$ will be denoted by $\psi_A^{(1)}(d_h^{(E)})$ and $\xi_A^{(1)}(d_h^{(E)})$ correspondingly. Similarly, the set of all second components will be denoted by $\psi_A^{(2)}(d_h^{(E)})$ and $\xi_A^{(2)}(d_h^{(E)})$.

Lemma 5. *For any diagonal* $d_h^{(E)} \in D^{(E)}$ $\psi_A(d_h^{(E)}) \neq \emptyset$.

Lemma 5 shows that if the transition function of a given automaton is completely defined then any essential diagonal can be reached during the execution of the algorithm.

Lemma 6. *If* $A_1 \sim A_2$ *then for any diagonal* $d_h^{(E)} \in D^{(E)}$
$\xi_{A_1}^{(1)}(d_h^{(E)}) = \xi_{A_2}^{(1)}(d_h^{(E)})$.

Lemma 7 below shows that if two automata are not equivalent then there exists a diagonal (not necessary essential) and a cell on that diagonal such that either both automata reach the cell - one in a final state and the other not in a final state or one automata reaches the cell and the second does not reach it.

Lemma 7. *If* $A_1 \not\sim A_2$ *then there exists a diagonal* d *and a cell* a_j, $a_j \in U_d^{(A_j)}$, $j \in \{1, 2\}$ *and a number* j', $j' \neq j$, $j' \in \{1, 2\}$ *that* $\phi_{A_j}(a_j)$ *is defined,* $\phi_{A_j}(a_j) \in F_{A_j}$ *and* $\phi_{A_{j'}}(a_j)$ *is not defined or* $\phi_{A_{j'}}(a_j)$ *is defined, but* $\phi_{A_{j'}}(a_j) \notin F_{A_{j'}}$.

Lemma 8. *For any number* $m > k$ *there exists a number* $m' < m$, $m'' \leq m$, $m' < m''$, $d_{m'}^{(E)}$, $d_{m''}^{(E)}$ *are essential diagonals* $\psi_{A_1}^{(2)}(d_{m'}^{(E)}) = \psi_{A_1}^{(2)}(d_{m''}^{(E)})$ *and* $\psi_{A_2}^{(2)}(d_{m'}^{(E)}) = \psi_{A_2}^{(2)}(d_{m''}^{(E)})$.

Lemma 9 shows that if there exist two essential diagonals with repeating set of states then it is possible to determine next essential diagonal with the same set of states. The distance between diagonals is not constant, but it is computable.

Lemma 9. *$\forall m' < k$, $m'' \leq k$, $m' < m''$ $d_{m'}^{(E)}$, $d_{m''}^{(E)}$ are essential diagonals and $\psi_A^{(2)}(d_{m'}^{(E)}) = \psi_A^{(2)}(d_{m''}^{(E)})$ then $\psi_A^{(2)}(d_{m''}^{(E)}) = \psi_A^{(2)}(d_{m'''}^{(E)})$ $m''' = m'' + 2^{m''} + \ldots + 2^{m''+(m''-m')-1}$.*

Lemma 10. *Let $p = a_{p_1} \ldots a_{p_m}$, $m > k$ be a complete path for the automaton A. Let also $d_{m_1}^{(E)}, \ldots, d_{m_u}^{(E)}$, $m_1 < \ldots < m_u < m$, $u \geq |S|$ be a set of essential diagonals such that $\psi_A^{(2)}(d_{m_t}^{(E)}) = \psi_A^{(2)}(d_{m_1}^{(E)})$ $t = 1, \ldots, u$. Then there exist numbers m', $m'' \in \{m_1, \ldots, m_u\}$ such that $\phi_A(a_{p_{m'}}) = \phi_A(a_{p_{m''}})$, $a_{p_{m'}} \in U_{d_{m'}^{(E)}}$, $a_{p_{m''}} \in U_{d_{m''}^{(E)}}$.*

Lemma 11 below shows that if the set of states is repeating then it is possible to find essential diagonals with repeating states for each state of the set.

Lemma 11. *Let $d_{m'}^{(E)}$, $d_{m''}^{(E)}$, $m'' > m'$ be essential diagonals, $a_{p_{m'}} \in U_{d_{m'}^{(E)}}$, $a_{p_{m''}} \in U_{d_{m''}^{(E)}}$ $\psi_A^{(2)}(d_{m'}^{(E)}) = \psi_A^{(2)}(d_{m''}^{(E)})$. Then for any $s \in \psi_A^{(2)}(d_{m'})$ there exists a number $m''' \geq m''$ and there exists a cell $a_{p_{m'''}} \in U_{d_{m'''}^{(E)}}$ $\phi_A(a_{p_{m'}}) = \phi_A(a_{p_{m'''}}) = s$.*

If $a_1, a_2 \in N^n$ then the distance between a_1 and a_2 will be denoted by $dist(a_1, a_2)$.
Lemma 12 extends Lemma 11 for the case when there are two automata.

Lemma 12. *Let $p_j = a_{p_{j1}} \ldots a_{p_{jm}}$, $m > k$ be a complete path for the automaton A_j, $j = 1, 2$. Let also $d_{m_1}^{(E)}, \ldots, d_{m_u}^{(E)}$, $m_1 < \ldots < m_u < m$ be a set of diagonals such that $\psi_{A_j}^{(2)}(d_{m_t}^{(E)}) = \psi_{A_j}^{(2)}(d_{m_1}^{(E)})$, $t = 1, \ldots, u$, $u < |S|$, $j = 1, 2$. Then there exist numbers m', $m'' \in \{m_1, \ldots, m_u\}$ such that $\phi_{A_j}(a_{p_{m'}}) = \phi_{A_j}(a_{p_{m''}})$, $a_{p_{jm'}} \in U_{d_{m'}^{(E)}}$, $a_{p_{jm''}} \in U_{d_{m''}^{(E)}}$.*

Lemma 13 shows that if

1) for equivalent automata there are two paths accepted by each automaton with states repeating on same essential diagonals;
2) these paths end in the same cell;
then the distance between cells reached on diagonals with repeating states does not change.

Lemma 13. *Let $p_j = a_{p_{j1}} \ldots a_{p_{jm}}$, $m > k$ be an accepted path for the automaton A_j, $j = 1, 2$, $d_{m'}^{(E)}$, $d_{m''}^{(E)}$, $m' < m'' < m$ be essential diagonals, $a_{p_{jm'}} \in U_{d_{m'}^{(E)}}$, $a_{p_{jm''}} \in U_{d_{m''}^{(E)}}$, $\psi_{A_j}^{(2)}(d_{m'}^{(E)}) = \psi_{A_j}^{(2)}(d_{m''}^{(E)})$, $\phi_{A_j}(a_{p_{jm'}}) = \phi_{A_j}(a_{p_{jm''}})$. Then $A_1 \sim A_2$ and $a_{p_{1m}} = a_{p_{2m}} = a_{p_m}$ implies that $dist(a_{p_{1m'}}, a_{p_{2m'}}) = dist(a_{p_{1m''}}, a_{p_{2m''}})$.*

Proof. Assume that $dist(a_{p_{1m''}}, a_{p_{2m''}}) > dist(a_{p_{1m'}}, a_{p_{2m'}})$. Without additional restrictions it can be assumed that the number of essential diagonals $d_{m''}^{(E)}, \ldots, d_m^{(E)}$ containing cells $a_{p_{jm''}}, \ldots, a_{p_{jm}}$ is less than $max\{|S_1|, |S_2|\}$.

Due to the made assumption on difference in cells, paths that start from $a_{p_{1m'}}$ and $a_{p_{2m'}}$ which have the same characteristics as the characteristics of the path, are ending in different cells denoted further by $a'_{p_{1m}}$ and $a'_{p_{2m}}$ correspondingly. $\phi_{A_j}(a'_{p_{jm}})$ belongs to the set of final states of the automaton A_j, $j = 1, 2$. Meantime, $\phi_{A_1}(a'_{p_{2m}})$ and $\phi_{A_2}(a'_{p_{1m}})$ do not belong to the set of final states of A_1 and A_2, correspondingly, due to $A_1 \sim A_2$. Otherwise, returning back to the cell a_{p_m} it is evident that then the automaton A_j, $j = 1, 2$, will have two different accepted paths ending in the same cell which contradicts to the assumption that A_j is deterministic.

Lemma 14. $CF_{AP_{A_1}(\omega_1)} = CF_{AP_{A_2}(\omega_2)} \Longleftrightarrow A_1 \sim A_2$.

Proof. First, it will be shown that $CF_{AP_{A_1}(\omega_1)} = CF_{AP_{A_2}(\omega_2)} \Longrightarrow A_1 \sim A_2$.

Assume, that A_1 is not equivalent to A_2 and there are no paths p_1 and p_2 with the same canonic form in the trace words ω_1 and ω_2 that p_1 is accepted by A_1 and p_2 is not accepted by A_2 or, vice versa, p_2 is accepted by A_2 and p_1 is not accepted by A_1.

Without reducing the assumption and due to Lemma 7 there exists a complete path of length m, $m > k$ $p^{(j)} = a^{(j)}_{p_1} \ldots a^{(j)}_{p_m}$, $a^{(1)}_{p_m} = a^{(2)}_{p_m} = a_{p_m}$, for the automaton A_j, $j = 1, 2$ such that $\phi_{A_1}(a_{p_m})$ belongs to the set of final states of the automaton A_1 and $\phi_{A_2}(a_{p_m})$ does not belong to the set of final states of the automaton A_2 or $\phi_{A_2}(a_{p_m})$ is not defined.

The case when $\phi_{A_2}(a_{p_m})$ is defined will be considered at first.

As $m > k$ there exist, according to Lemma 8, essential diagonals with numbers m', m'', $m' < m$, $m'' < m$, $m' < m''$ such that $\psi^{(2)}_{A_1}(d^{(E)}_{m'}) = \psi^{(2)}_{A_1}(d^{(E)}_{m''})$ and $\psi^{(2)}_{A_2}(d^{(E)}_{m'}) = \psi^{(2)}_{A_2}(d^{(E)}_{m''})$.

Thus, per Lemma 11, a path $p^{(j)}$, $j = 1, 2$ can be represented in the following way: $p^{(j)} = a^{(j)}_{p_1} \ldots a^{(j)}_{p_{m'}} \ldots a^{(j)}_{p_{m''}} \ldots a_{p_m}$, where $\phi_{A_1}(a^{(1)}_{p_{m'}}) = \phi_{A_1}(a^{(1)}_{p_{m''}})$ and $\phi_{A_2}(a^{(2)}_{p_{m'}}) = \phi_{A_2}(a^{(2)}_{p_{m''}})$. Without reducing the assumption it can be assumed that $p^{(j)}$ are shortest paths for which A_1 and A_2 are not equivalent.

Let χ_j be a characteristics of the subpath $a^{(j)}_{p_{m''}} \ldots a_{p_m}$ and $p^{(j)}_1 = a^{(j)}_{p_{m'}} \ldots a^{(j)}_p$ is a subpath which has the same characteristics χ_j, $j = 1, 2$.

It is evident that such a subpath exists. Due to Lemma 13 $a^{(1)}_p = a^{(2)}_p = a_p$. Denote by $p^{(j)}_2$ the subpath of the path $p^{(j)}$, starting from a_{p_1} and ending with a predecessor of $a^{(j)}_{m'}$. Consider as a new path the following concatenation of those subpaths $p^{(j)}_{new} = p^{(j)}_2 p^{(j)}_1$. Its length is less than the length of the initial path $p^{(j)}$. Meantime, it implies evidently that $\phi_{A_1}(a_p)$ is a final state of A_1, but $\phi_{A_2}(a_p)$ is not a final state of A_2. Meantime, according to Lemmas 12, 13 there is no other complete path for the automaton A_2 to the cell a_p, for which $\phi_{A_2}(a_p)$ belongs to the set of final states. Thus, we obtain that for the diagonal with the number equal to the length of $p^{(j)}_{new}$ there exists a cell for which one of automata has a state different from final states. If the length of $p^{(j)}_{new}$ is still more than k, similar considerations should be done until the obtained paths have a length not exceeding k. Then having these paths we come to a contradiction with the initial assumption of the Lemma 14.

Now the case when $\phi_{A_2}(a_{p_m})$ is not defined is considered.

Similarly to the considerations above one may obtain that $\phi_{A_2}(a_p)$ is also not defined. Thus, it is obtained that there exists a cell at diagonal with the number equal to the length of $p_{new}^{(1)}$ for which ϕ_{A_1} is defined and its value belongs to the set of final states for A_1 and, at the same time, ϕ_{A_2} is not defined.

If the number of the diagonal is still more than k, similar considerations shall be done until the obtained path has a length not exceeding k. Then, having the path and ϕ_2 not defined at the end cell of the path we come to a contradiction with the assumption of Lemma 14.

As $A_1 \sim A_2 \Longrightarrow CF_{AP_{A_1}(\omega_1)} = CF_{AP_{A_2}(\omega_2)}$ is obviously also true, the Lemma is proved.

Theorem 1. *The equivalence problem of deterministic multitape automata is solvable.*

References

1. Bird, M.: The equivalence problem for deterministic two-tape automata (1973)
2. Clifford, A.H., Preston, G.B.: The algebraic theory of semigroups (1961)
3. Godlevskii, A.B., Letichevskii, A.A., Shukuryan, S.K.: Reducibility of program-scheme functional equivalence on a nondegenerate basis of rank unity to the equivalence of automata with multidimensional tapes (1980)
4. Grigorian, H.A., Shoukourian, S.K.: The equivalence problem of multidimensional multitape automata (2008)
5. Harju, T., Karhumäki, J.: The equivalence problem of multitape finite automata (1991)
6. Rabin, M.O., Scott, D.: Finite automata and their decision problems (1959)
7. Shoukourian, A.S.: Equivalence of regular expressions over a partially commutative alphabet (2009)