

# An approach to the classification of the loops of finite automata. Part II: The classification of the states based on the loops

B. F. Melnikov, A. A. Melnikova

**Abstract**—In this paper, we considered questions of the possible classification of the states and loops of a nondeterministic finite automaton.

For the development of algorithms for equivalent transformation of nondeterministic finite automata, we consider the basis finite automaton for the given regular language and the paths and loops of its transition graph. We also consider the paths and loops of the transition graph of another nondeterministic automaton that defines the same language. On the basis of this, we define corresponding paths and loops of two mentioned automata and the questions of their classification. This classification gives, for example, the possibilities for describing some heuristic algorithms for minimization of nondeterministic automata.

For the last thing, we describe the following objects. For each state of the basis automaton, we consider the states of the given automaton corresponding to this state of the basis automaton, and give their classification as a function of the loops passing through the same state of the basis automaton. Their subset is the set of so-called including loops, on the basis of which we determine so-called partially complete loops. For any chosen vertex of the basic automaton, we call the vertices of the considered nondeterministic automaton, through which all possible partially complete loops pass, by complete cyclic states.

At the end of the paper, we formulate the hypothesis that if for any state of the considered nondeterministic automaton, there exists at least one corresponding state of the basis automaton, such that the first one is a complete cyclic state for the second one, then all the corresponding states of the basis automaton are such ones.

In the presented Part II of the paper, we are considering issues of a special classification of states of any nondeterministic finite automaton. This classification is based on the application of developed in the first part of this paper classification of the loops.

**Keywords**—nondeterministic finite automata, basis automaton, transition graph, path, loop, algorithms for equivalent transformation, universal automaton.

This paper is a continuation of [1] (i.e., Part I). We consider the issues of a special classification of states of any nondeterministic finite automaton. This classification is based on the application of developed in Part I of this paper classification of the loops.

We continue the numeration of sections, definitions, figures, equations, etc. For the convenience of reading, we shall also repeat some figures and tables given already in [1], [2].

Received September 5, 2018.  
Boris F. Melnikov, Russian State Social University (email: bf-melnikov@yandex.ru).  
Aleksandra A. Melnikova, National Research Nuclear University “MEPhI” (email: super-avahi@yandex.ru).

For the designations of automata and its states on the figures, see the agreement of [3, Page 3]. (We used this agreement also in all previous papers.)

## VII. SETS $\Psi(q)$ , GRIDS AND PSEUDO-GRIDS

**Definition 2:** For each state  $q \in Q$ , we shall denote

$$\Psi(q) = \varphi_K^{in}(q) \times \varphi_K^{out}(q). \quad \square$$

In fact, this definition also defines a pseudo-grid; definitions and examples of pseudo-grids and grids were given in [4, Section 3].

Let us continue to consider regular language

$$(a + ab + ba)^*. \quad (9)$$

(i.e., [1, Eq. (5)]). We continue to consider two the following automata for this language:

- the canonical automaton of Fig. 12 (i.e., [1, Fig. 4]):

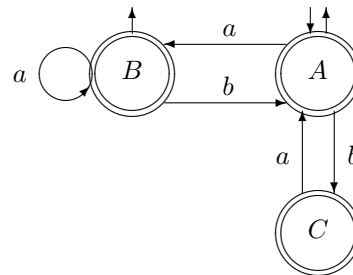


Fig. 12

- automaton of Fig. 13 (i.e., [1, Fig. 6]):

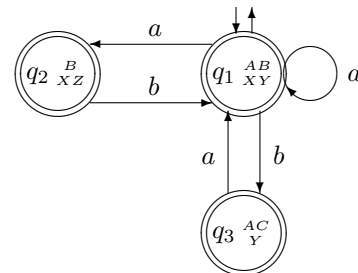


Fig. 13

We shall also consider a new automaton for the same language (9):

- automaton of Fig. 14, see below; for the considered language  $L$ , it is the *universal automaton* (see [5], [6], etc.), or automaton  $COM(L)$  (see [7], [8], etc.).

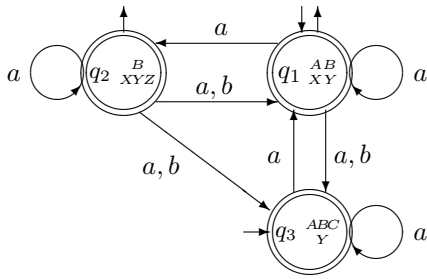


Fig. 14

As we said in [1], this language coincides with its mirror image, therefore the canonical automaton for its mirror image can be obtained replacing  $A, B$  and  $C$  for  $X, Y$  and  $Z$  respectively. Also like [1], [9], we marked values of state-marking functions for each state of the both non-canonical automata in abbreviated form: e.g., for automaton on Fig. 14, we have

$$\varphi^{out}(q_2) = \{X, Y, Z\}.$$

(Like before, these values can be obtained by the simple algorithms of [4, Section 3].)

Continuing the calculations, we obtain that the following facts hold:

- state  $q_1$  of the first automaton and all three states of the second automaton form grids;
- states  $q_2$  and  $q_3$  of the first automaton form pseudo-grids.

### VIII. CORRESPONDING LOOPS AND $\overset{A}{X}$ -CYCLIC STATES

**Definition 3:** Let for some states

$$q \in Q, \quad A \in Q_\pi \quad \text{and} \quad X \in Q_\rho$$

be

$$(A, X) \in \Psi(q).$$

Let there exist a  $\overset{A}{X}$ -loop of automaton  $\mathcal{BA}(L)$  having corresponding long loop of automaton

$$K = (Q, \Sigma, \delta, S, F)$$

[1, Eq. (1)], passing  $q$ ; moreover, let states

$$\overset{A}{X} \quad \text{and} \quad q$$

be corresponding in these loops. Then we shall call  $q$  by  $\overset{A}{X}$ -cyclic state.  $\square$

For the last definition, there are very important two the following remarks.

- We do not require that this  $\overset{A}{X}$ -loop is simple.
- $\Psi(q)$  may contain not only the pair  $(A, X)$ . Therefore in the considered loops,  $q$  may also correspond to another state.

Let us consider some examples. For the same language (9), let us first consider its automaton  $\mathcal{BA}(L)$ , Fig. 15; this is automaton of [1, Fig. 5]. Let us also consider the “modified basis automaton” having a long loop (2-loop), Fig. 16; this is automaton of [1, Fig. 11]. It is simply to obtain, that the state-marking functions of automaton on Fig. 16 are also shown on that figure; e.g.,

$$\varphi^{out}(\overset{B}{Y}1) = Y,$$

etc.

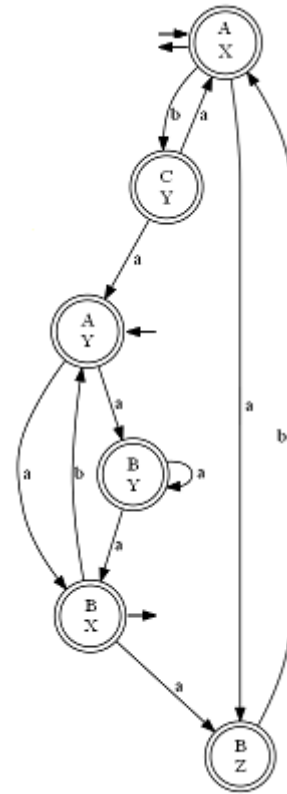


Fig. 15

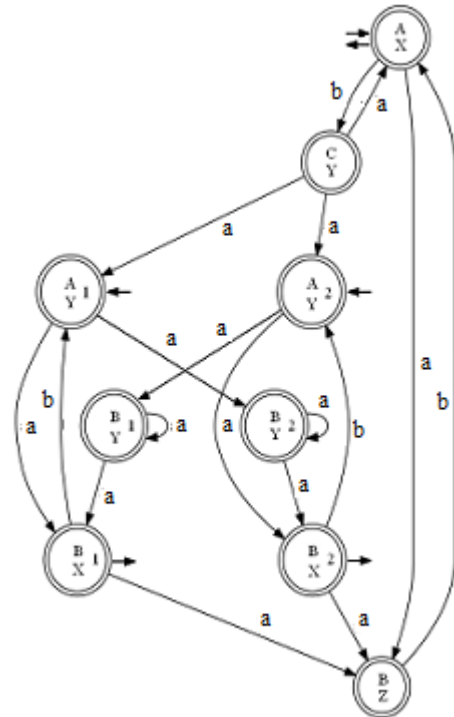


Fig. 16

For the transition graph of automaton on Fig. 16, we have no loop corresponding to the loop

$$\overset{B}{X} \xrightarrow{b} \overset{A}{Y} \xrightarrow{a} \overset{B}{Y} \xrightarrow{a} \overset{B}{X}$$

of  $\mathcal{BA}(L)$ . However, the last automaton includes corresponding long loop; that is

$$\overset{B}{X}1 \xrightarrow{b} \overset{A}{Y}1 \xrightarrow{a} \overset{B}{Y}2 \xrightarrow{a} \overset{B}{X}2 \xrightarrow{b} \overset{A}{Y}2 \xrightarrow{a} \overset{B}{Y}1 \xrightarrow{a} \overset{B}{X}1.$$

E.g.,  $\overset{A}{Y}2$  is  $\overset{A}{Y}$ -cyclic state.

Thus, we think that Definition 3 is simple. Then in following examples, we shall not specify cyclic states.

In more complicated examples, there exist possibilities, when, informally speaking, properties of the both considered examples hold. For example:

- the considered automaton *cannot* be obtained from the basis one by “duplicating” its loops;
- for a loop of the basis automaton, there exists corresponding  $n$ -loop, but only for  $n > 1$ .

For this thing, let us consider the following examples (Fig. 17 and 18). These automata accept the same language. (As before, we show the values of state-marking functions.)

Consider simple loop

$$\tilde{\nu} = \frac{A}{Y} \xrightarrow{a} \frac{B}{X} \xrightarrow{b} \frac{A}{Y} \quad (10)$$

of automaton  $\mathcal{BA}(L)$ . Automaton on Fig. 17...

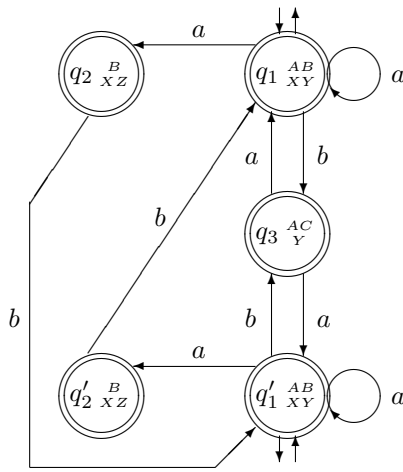


Fig. 17

... has corresponding 2-loop

$$q_1 \xrightarrow{a} q_2 \xrightarrow{b} q'_1 \xrightarrow{a} q'_2 \xrightarrow{b} q_1 \quad (11)$$

(and has no corresponding 1-loops). And automaton on Fig. 18 (on this figure, we marked the added edges by the bold font) ...

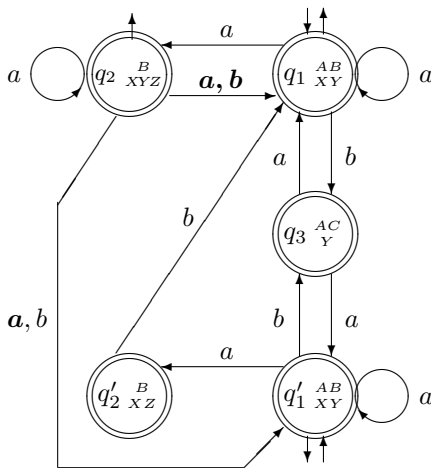


Fig. 18

... has the same corresponding loop (11), and also corresponding 1-loop

$$q_1 \xrightarrow{a} q_2 \xrightarrow{b} q_1.$$

But there exists examples, when  $(A, X) \in \Psi(q)$ , but  $q$  is *not*  $\frac{A}{X}$ -cyclic state. Let us consider the following modification of automaton on Fig. 17 (i.e., Fig. 17'):

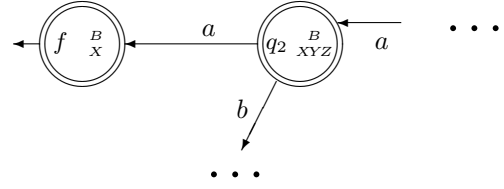


Fig. 17'

(For this automaton, we added a new final state  $f$  and the only edge

$$q_2 \xrightarrow{a} f;$$

all the other states and edges are the same. We can simply obtain, that

$$\varphi^{in}(f) = \{B\} \quad \text{and} \quad \varphi^{out}(f) = \{X\}.$$

The automaton defines the same language.)

Let us remark, that values of  $\varphi^{in}$  and  $\varphi^{out}$  for *all the other states do not change*. We show these values for states  $q_2$  and  $f$  only; it is important to remark, that value  $\varphi^{out}(q_2)$  changes. Thus,

$$\varphi^{out}(q_2) \ni Y$$

for obtained automaton, then

$$(B, Y) \in \Psi(q_2).$$

But it is simple to prove, that  $q_2$  is not  $\frac{B}{Y}$ -cyclic state.

### IX. ONCE MORE ABOUT WATERLOO LANGUAGE: LONG LOOPS OF ITS BASIS AUTOMATON

Let us consider also so-called automaton Waterloo, [10]. The figures and tables of this section are given by [2]; about related problems, see also [5], [11], [12].

The given automaton is usually defined by the following transition graph, Fig. 19:

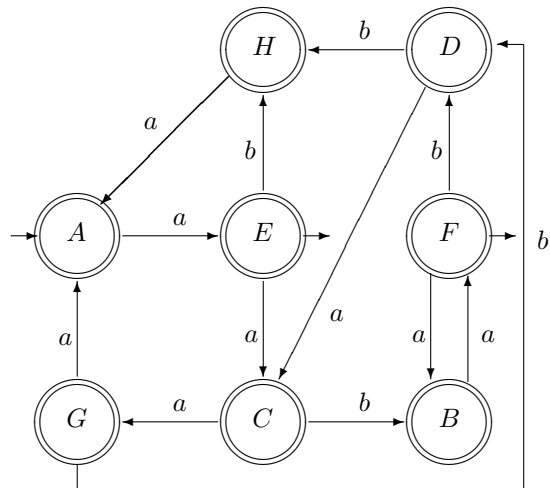


Fig. 19

In this section, we shall denote its language by  $L$ . For it<sup>1</sup>, the following tables describe the equivalent basis automaton  $\mathcal{BA}(L) \dots$

Tab. 4

| $\mathcal{BA}(L)$  | $a$          | $b$          |
|--------------------|--------------|--------------|
| $\rightarrow A\#Y$ | $E\#X, E\#P$ | -            |
| $\rightarrow A\#Q$ | $E\#W$       | -            |
| $B\#Y$             | $F\#X, F\#P$ | -            |
| $B\#V$             | $F\#Z, F\#R$ | -            |
| $C\#U$             | $G\#S$       | $B\#Y, B\#V$ |
| $C\#V$             | $G\#Z, G\#R$ | -            |
| $D\#W$             | $C\#U$       | $H\#Z, H\#S$ |
| $D\#P$             | $C\#V$       | -            |
| $\leftarrow E\#X$  | -            | -            |
| $E\#W$             | $C\#U$       | $H\#Z, H\#S$ |
| $E\#P$             | $C\#V$       | -            |
| $\leftarrow F\#X$  | -            | -            |
| $F\#Z$             | $B\#Y$       | -            |
| $F\#P$             | $B\#V$       | -            |
| $F\#R$             | -            | $D\#W, D\#P$ |
| $G\#Z$             | $A\#Y$       | -            |
| $G\#R$             | -            | $D\#W, D\#P$ |
| $G\#S$             | $A\#Q$       | -            |
| $H\#Z$             | $A\#Y$       | -            |
| $H\#S$             | $A\#Q$       | -            |

$\dots$  and the equivalent universal automaton  $\mathcal{COM}(L)$ :

Tab. 5

| $\mathcal{COM}(L)$ | $a$              | $b$     |
|--------------------|------------------|---------|
| $\rightarrow 1$    | 6, 7, 8, 14      | -       |
| $\rightarrow 2$    | 8, 14            | -       |
| 3                  | 8, 9, 10, 13, 14 | -       |
| 4                  | 10, 13           | -       |
| 5                  | 10, 11, 12, 13   | 2, 3, 4 |
| 6                  | 4, 5             | 12, 13  |
| $\leftarrow 7$     | 4, 5             | 12, 13  |
| $\leftarrow 8$     | 4                | -       |
| $\leftarrow 9$     | 2, 3, 4          | 6, 14   |
| 10                 | 2                | 6, 14   |
| 11                 | 1, 2             | 6, 14   |
| 12                 | 1, 2             | -       |
| 13                 | 2                | -       |
| 14                 | 4                | -       |

See details, e.g., in [2]; and in [1, Sect.II], we gave the agreement about of the states of basis automata, i.e., of their notation in the tables.

On the basis of these tables, we make the following constructions. Let us consider the loop

$$C \xrightarrow{a} G \xrightarrow{b} D \xrightarrow{a} V$$

of the basis automaton. For the specified automata and loop, there exist the following *different* corresponding 1-loops

$$5 \xrightarrow{a} 11 \xrightarrow{b} 6 \xrightarrow{a} 5 \quad \text{and} \quad 4 \xrightarrow{a} 10 \xrightarrow{b} 14 \xrightarrow{a} 4,$$

and also corresponding 2-loop

$$4 \xrightarrow{a} 10 \xrightarrow{b} 6 \xrightarrow{a} 5 \xrightarrow{a} 11 \xrightarrow{b} 14 \xrightarrow{a} 4.$$

Let us remark, that the last loop is simple.

<sup>1</sup> And also for 10 states of the canonical automaton of language  $L^R$ :

$X$  (the initial state),  $Y, Z, U, V, W, P, Q, R, S$ .

We usually denote such an automaton by  $\widetilde{L}^R$ . But we shall not use automaton  $\widetilde{L}^R$  in this paper (we shall use its states only).

The states of automaton  $\mathcal{COM}(L)$  are the numbers from 1 to 14.

### X. $\psi$ -PROPER LOOPS

**Definition 4:** Let some

$$\alpha \subseteq Q_\pi, \quad \beta \subseteq Q_\rho \quad \text{and} \quad \psi = \alpha \times \beta.$$

Let us consider a loop  $\tilde{\nu}$  of automaton  $\mathcal{BA}(L)$ . Let there exist a corresponding long loop  $\nu$  of automaton

$$K = (Q, \Sigma, \delta, S, F)$$

[1, Eq. (1)] containing a state  $q$ , such that

$$\Psi(q) = \psi.$$

Then we shall call  $\tilde{\nu}$  by  $\psi$ -proper loop.

For it, all such its states  $q$  (i.e., the states of the considered type) are called  $(\tilde{\nu}, \psi)$ -proper states.  $\square$

The following remark is very important, it is illustrated by the examples considered below. Usually, state  $q$  is a priori given, and  $\psi = \Psi(q)$ . If so, then we allow both the cases

$$q' = q \quad \text{and} \quad q' \neq q.$$

Therefore, e.g., an  $\Psi(q)$ -proper loop may contain and may not contain  $q$ .

As we said before, automaton on Fig. 17 has corresponding 2-loop for loop (10). But, certainly, (10) is

$$\text{a } \{B\} \times \{X, Z\}\text{-proper } \overset{A}{Y}\text{-loop}$$

(for this automaton).

Similarly, for automaton on Fig. 18, loop (10) is

$$\text{a } \{B\} \times \{X, Z\}\text{-proper } \overset{A}{Y}\text{-loop.}$$

Also, for automaton on Fig. 18, the same loop is

$$\text{a } \{B\} \times \{X, Y, Z\}\text{-proper } \overset{A}{Y}\text{-loop}$$

and

$$\text{a } \{A, B\} \times \{X, Y\}\text{-proper } \overset{A}{Y}\text{-loop.}$$

### XI. $(\overset{A}{X}, q)$ -PARTIALLY COMPLETE LOOPS

The examples of the previous section lead to the need to consider the following definition.

**Definition 5:** Let:

- $q$  be  $\overset{A}{X}$ -cyclic state;
- $\tilde{\nu}$  be *some*  $\Psi(q)$ -proper  $\overset{A}{X}$ -loop of  $\mathcal{BA}(L)$  (remark that, therefore,  $q$  is a  $(\tilde{\nu}, \Psi(q))$ -proper state);
- there exists corresponding long  $q$ -loop  $\nu$  of automaton (1), which contains *no other*  $(\tilde{\nu}, \Psi(q))$ -proper state.<sup>2</sup>

Then we shall call  $\tilde{\nu}$  by  $(\overset{A}{X}, q)$ -partially complete loop.  $\square$

Let us continue to consider previous examples. For both automata (on Fig. 17 and Fig. 18), we consider state  $q_1$  as  $\overset{A}{Y}$ -cyclic state and loop (10).

Both automata on Fig. 17 and 17' have the only corresponding long  $q_1$ -loop, i.e., (11); besides, (11) contains another state (i.e.,  $q'_1$ ), which is also

$$((10), \Psi(q_1))\text{-proper state.}$$

Then for these automata, loop (10) is *not*  $(\overset{A}{Y}, q_1)$ -partially complete loop.

And automaton on Fig. 18 has another corresponding long  $q_1$ -loop, i.e.,

$$q_1 \xrightarrow{a} q_2 \xrightarrow{b} q_1.$$

<sup>2</sup> Here, we use not the terminology that we have introduced, but the usual terms of the graph theory. Then  $q$ -loop is simply a loop which includes state  $q$ . Notice that we do *not* require, that this long  $q$ -loop is 1-loop.

(See Part I of this paper about  $n$ -loops for  $n \geq 1$ .)

The last loop contains *no other*

$((10), \Psi(q_1))$ -proper states

$(q_1$  only), therefore it is a  $(\frac{A}{Y}, q_1)$ -partially complete loop. Quite similarly, the considered loop (10):

- is *not*  $(\frac{B}{X}, q_2)$ -partially complete loop for automaton on Fig. 17;
- and *is*  $(\frac{B}{X}, q_2)$ -partially complete loop for automaton on Fig. 18.

However,

- it is also a  $(\frac{B}{X}, q_2)$ -partially complete loop for automaton on Fig. 17', because for this automaton,  $\Psi(q_2) \neq \Psi(q'_2)$ .

## XII. $\frac{A}{X}$ -CYCLIC STATES

**Definition 6:** Let:

- $q$  be  $\frac{A}{X}$ -cyclic state; *each*
- $\Psi(q)$ -proper  $\frac{A}{X}$ -loop of automaton  $\mathcal{BA}(L)$  be  $(\frac{A}{X}, q)$ -partially complete loop.

Then we shall call  $q$  by **complete  $\frac{A}{X}$ -cyclic state**.  $\square$

The automata of previous subsections gives only some *simple* examples of complete cyclic states. E.g., for automaton on Fig. 17, we obtain complete cyclic states only for  $q_3$ : otherwise (i.e., for another  $q \neq q_3$  and *some* selected state  $\frac{A}{X}$  of the equivalent basis automaton), we can specify a loop which is not  $(\frac{A}{X}, q)$ -partially complete; then each  $q \neq q_3$  is not  $(\frac{A}{X}, q)$ -cyclic complete. Generally, for considered automata, there are complete states only if there are no other states with the same value  $\Psi$ .

Let us consider the last example, i.e., automaton on Fig. 20 (compare Fig. 18):

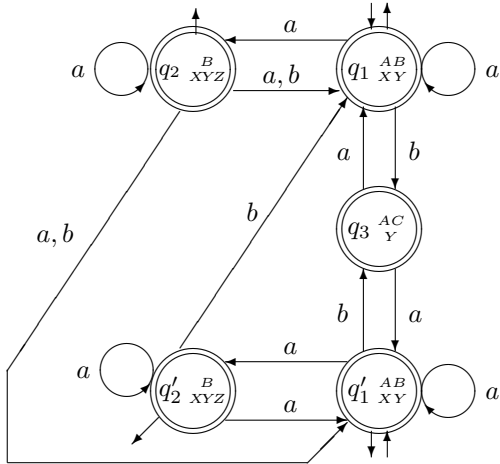


Fig. 20

This automaton also defines the same language (9). Then, like all the examples considered before, we can use the above automata  $\mathcal{BA}(L)$  and  $\mathcal{COM}(L)$ , and also their loops.

For the last automaton of Fig. 20, let us consider the loops corresponding to *simple*  $\frac{B}{X}$ -loops (or their powers) of automaton  $\mathcal{BA}(L)$ , where  $\frac{B}{X}$  corresponds to  $q'_2$ . The loop

$$\frac{B}{X} \xrightarrow{b} \frac{A}{Y} \xrightarrow{a} \frac{B}{Y} \xrightarrow{a} \frac{B}{X}$$

(or, rather,  $\frac{B}{X} \xrightarrow{b} \frac{A}{Y} \xrightarrow{a} \frac{B}{Y} \xrightarrow{a} \frac{B}{X} \xrightarrow{b} \frac{A}{Y} \xrightarrow{a} \frac{B}{Y} \xrightarrow{a} \frac{B}{X}$ )

corresponds, for example, to

$$q'_2 \xrightarrow{b} q_1 \xrightarrow{a} q_1 \xrightarrow{a} q_1 \xrightarrow{b} q_3 \xrightarrow{a} q'_1 \xrightarrow{a} q'_2.$$

And the loop

$$\frac{B}{X} \xrightarrow{a} \frac{B}{Z} \xrightarrow{b} \frac{A}{X} \xrightarrow{b} \frac{C}{Y} \xrightarrow{a} \frac{A}{Y} \xrightarrow{a} \frac{B}{X}$$

corresponds, for example, to

$$q'_2 \xrightarrow{a} q'_2 \xrightarrow{b} q_1 \xrightarrow{b} q_3 \xrightarrow{a} q'_1 \xrightarrow{a} q'_2.$$

Thus, both the considered loops are

$$\{B\} \times \{X, Y, Z\}\text{-proper } \frac{B}{X}\text{-loops,}$$

because:

- the considered corresponding loops of automaton on Fig. 20 do not contain  $q_2$ ;
- besides,  $q_2$  is the only other state of this automaton, such that  $\Psi(q_2) = \Psi(q'_2)$ .

However, there is simple to prove, that the loop

$$\frac{B}{X} \xrightarrow{b} \frac{A}{Y} \xrightarrow{a} \frac{B}{X} \quad (12)$$

is *not*  $\{B\} \times \{X, Y, Z\}$ -proper. Then state  $q'_2$  is *not* complete  $\frac{B}{X}$ -cyclic one.

## XIII. SETS $\hat{\Psi}(q)$ AND AN IMPORTANT HYPOTHESIS

The examples of the previous section lead to the need to consider the following definition.

**Definition 7:**

$$\hat{\Psi}(q) = \{ (A, X) \in \Psi(q) \mid q \text{ is complete } \frac{A}{X}\text{-cyclic state} \}.$$

$\square$

For each of the considered examples, the set  $\hat{\Psi}(q)$  is either empty or equal to  $\Psi(q)$ . But, as we said before, we cannot give *interesting* examples (i.e., when for the considered language  $\Psi(q) \neq \hat{\Psi}(q)$ ). The *informal* explanation of this fact is the following.

- Automaton  $\mathcal{BA}(L)$  for this language has the only strongly connected component (see [13], [14], etc.).
- Let us suppose, that there exists a pair of states  $(\frac{A}{X}, q)$  for which  $q$  is complete  $(\frac{A}{X}, q)$ -cyclic, then there exists a loop of  $\mathcal{BA}(L)$  which is not complete  $(\frac{A}{X}, q)$ -cyclic. Let this loop be  $\hat{\nu}$ .
- Then (owing to the only strongly connected component) we can consider  $\hat{\nu}$  as a part of some  $\frac{B}{X}$ -loop for *any* state  $\frac{B}{X}$ , where  $\frac{B}{X}$  may be absent in  $\hat{\nu}$ . (Because we consider not only simple loops, see before.)

Thus, automata  $\mathcal{BA}(L)$  have (in our examples) the only strongly connected components. We shall not consider other examples (i.e., automata  $\mathcal{BA}(L)$  having 2 or more strongly connected components) because such examples are heavy. Omitting the detailed explanation, we already considered *the language*  $L$ , for which automaton  $\mathcal{BA}(L)$  has 4 strongly connected components; this was the language defined by automaton of [1, Fig. 1] (or [4, Fig. 1]). However, each of the other automata usually considered by us (i.e., automata

$$\tilde{L}, \tilde{L}^R, (\tilde{L}^R)^R, \mathcal{COM}(L)$$

for its language  $L$  has 1 strongly connected component only. This is easily verified from the tables of [1], [4].

Let us formulate the above of this section in the form of the following hypothesis.

**Hypothesis 1:** For each state of any nondeterministic finite automaton

$$K = (Q, \Sigma, \delta, S, F)$$

[1, Eq. (1)], one of the following two equalities holds:

- either  $\hat{\Psi}(q) = \emptyset$ ;
- or  $\hat{\Psi}(q) = \Psi(q)$ . □

#### XIV. CONCLUSION. POSSIBLE DIRECTIONS FOR THE FURTHER RESEARCH

Thus, we have considered a possible approach to classification states and loops of a nondeterministic finite automaton. Let us note once again that, that unlike [9] devoted to an approach to the classification of *languages*, in this paper we have considered *various nondeterministic finite automata* defining some given language. We believe that the description of such relationship (correspondence) between the loops of an nondeterministic automaton and the loops of the equivalent basis automaton makes it possible to solve both “purely theoretical”, and “purely practical” problems of the formal languages theory.

About “purely practical” problems, we have written in our previous publications, in particular, in papers cited in this one; moreover, some of them ([2], [7], [8]) are devoted directly to the practical application of the developed algorithms. We have in the first place the description of heuristic minimization algorithms by several different criteria. We think also, that many questions of the formal languages theory can be considered “located between theoretical and practical problems”.

We also are going to continue our previous publications (see also [15]) to describe the alternative approach to solving the star-height problem.<sup>3</sup> Very briefly, it can be described as follows.

After defining the star height for an automaton, considering all the possible orders of its states and making regular expressions for each order in the usual way, we show, that we can construct corresponding automaton for *each* regular expression, and therefore we can do this thing for *a hypothetical* regular expression defining the given regular language and having minimum possible star-height. Thus, there exists the minimum possible value of star-height for some hypothetical automaton defining the given regular language; let this automaton be  $K$ . We consider not only  $K$ , but also the concrete order  $\tau$  of its states defined in [15], corresponding to the regular expression having minimum possible star-height.

Considering the states of  $K$  in the order  $\tau$ , we obtain for the next state one of the three following things. Either each its loop has equivalent one which does not pass the considered state. Or there exists some other state, which has the smaller value of the order  $\tau$  and defines the same loops. Or we can add some edges to obtain one of previous cases.

Using a finite sequence of such steps, we obtain the automaton, which is equivalent to the given one; moreover, we can a priori limit the number of states of such “minimum”

automaton, using the knowledge of the given language only. Thus, this sequence of steps gives the nondeterministic finite automaton having a priori limited number of states, defining the given regular language and having the minimum possible star-height.

#### REFERENCES

- [1] Melnikov B. and Melnikova A. *An approach to the classification of the loops of finite automata. Part I: Long corresponding loops*. International Journal of Open Information Technologies. 2018, vol. 6, no. 9, pp. 9–14.
- [2] Melnikov B. and Tsyganov A. *The state minimization problem for nondeterministic finite automata: The parallel implementation of the truncated branch and bound method*. Proceedings of the International Symposium on Parallel Architectures, Algorithms and Programming, PAAP-2012. Taipei, Taiwan. 2012, pp. 194–201.
- [3] Melnikov B. and Melnikova A. *Pseudo-automata for generalized regular expressions*. International Journal of Open Information Technologies. 2018, vol. 6, no. 1, pp. 1–8.
- [4] Melnikov B. *Once more on the edge-minimization of nondeterministic finite automata and the connected problems*. Fundamenta Informaticae. 2010, vol. 104, no. 3, pp. 267–283.
- [5] Polák L. *Minimalizations of NFA Using the Universal Automaton*. International Journal of Foundations of Computer Science. 2004, vol. 16, no. 5, pp. 457–504.
- [6] Lombardy S. and Sakarovitch J. *The Universal Automaton*. Logic and Automata, Texts in Logic and Games, Amsterdam Univ. Press. 2008, vol. 2, pp. 457–504.
- [7] Dolgov V. and Melnikov B. *Postroenie universal'nogo konechnogo avtomata. I... [The construction of a universal finite automaton. Part I: From theory to practical algorithms]*. Vestnik of Voronezh State University. 2013, no. 2, pp. 173–181. (in Russian, <https://elibrary.ru/item.asp?id=20267924>)
- [8] Dolgov V. and Melnikov B. *Postroenie universal'nogo konechnogo avtomata. II... [The construction of a universal finite automaton. Part II: Examples of work of algorithms]*. Vestnik of Voronezh State University. 2014, no. 1, pp. 78–85. (in Russian, <https://elibrary.ru/item.asp?id=21445963>)
- [9] Melnikov B. *The complete finite automaton*. International Journal of Open Information Technologies. 2017, vol. 5, no. 10, pp. 9–17.
- [10] Kameda T. and Weiner P. *On the state minimization of nondeterministic finite automata*. IEEE Transactions on Computers. 1970, vol. C-19, no. 7, pp. 617–627.
- [11] Jiang T. and Ravikumar B. *Minimal NFA problems are hard*. SIAM Journal on Computing. 1993, vol. 22, no. 6, pp. 1117–1141.
- [12] Yo-Sub Han. *State elimination heuristics for short regular expressions*. Fundamenta Informaticae. 2013, vol. 128, pp. 445–462.
- [13] Harary F. *Graph Theory*. Addison Wesley (Boston), 1969, 274 p.
- [14] Gera R., Hedetniemi S., and Larson C., editors. *Graph Theory: Favorite Conjectures and Open Problems – 1*. Springer, 2016, 291 p.
- [15] Melnikov B. *The star-height of a finite automaton and some related questions*. International Journal of Open Information Technologies. 2018, vol. 6, no. 7, pp. 1–5.

Boris Feliksovich MELNIKOV,  
Professor of Russian State Social University  
(<http://www.rgsu.net/>),  
email: bf-melnikov@yandex.ru,  
mathnet.ru: personid=27967,  
elibrary.ru: authorid=15715,  
scopus.com: authorId=55954040300.

Aleksandra Aleksandrovna MELNIKOVA,  
Associated Professor (Docent)  
of National Research Nuclear University “MEPhI”  
(<https://mephi.ru/>),  
email: super-avahi@yandex.ru,  
elibrary.ru: authorid=143351.

<sup>3</sup> These two papers, as well as Part II presented here, may seem unrelated to each other. However, in their sections titled “Motivation”, we briefly described this relationship.