An approach to the classification of the loops of finite automata. Part I: Long corresponding 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 I of the paper, we consider the issues related to corresponding loops of the given nondeterministic automaton and equivalent basis automaton.

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

I. Introduction and motivation

We continue the series of papers of equivalent transformation of nondeterministic finite automata (NFAs); let us mention some of them: [1]–[7]. One (but not the only) goal of such publications is the description of algorithms for NFA-minimization. Speaking of this, we mean simultaneously some following dichotomies (trichotomies, etc.).

- The search for an equivalent automaton with a minimum number:
  - of *states*, see [8], [9] etc.;
  - of *edges*, see [2], [4] etc.;
  - or of the *star-height value*, see [10], [11] etc.

  More than that, it is possible to minimize by some *mixed* criteria; we are going to consider such options for minimization in one of the following publications.
- We consider both *exact* and *heuristic* algorithms; and among the last ones, we consider:
  - "usual" heuristic algorithms;
  - and so-called *anytime* algorithms, [12] etc.

It is important to note, that for all resulting classes of algorithms, we use the same auxiliary algorithms of equivalent transformation of NFA.

In this paper, we continue to consider the basis finite automaton for the given regular language, and also the paths and loops of its transition graph. We also consider the *corresponding* paths of the transition graph of *some* NFA that defines the given language.

Considered in this paper questions of the possible classification of the states and loops of a NFA give additional possibilities for describing algorithms mentioned before. For this thing, we consistently 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 basic automaton. We give their classification as a function of the loops passing through them corresponding to the same state of the basis automaton.
- Then we formulate two special *properties* of the automaton K related to all their states; for different NFAs defining the same regular language, these properties can either be fulfilled or not fulfilled.
- The special subset of loops of the given automaton corresponding to the same state of the basis automaton is the set of so-called *including loops*.
- On the basis of the last definition, 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 following hypothesis. If for any state of the considered NFA, there exists at least one corresponding state of the basic automaton, such that the first one is a complete cyclic state for the second one, then all the corresponding states of the basic automaton are such ones.

These objects and hypothesis are the main subject of this paper.

Received August 14, 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).

Like [11], the parts of the subject of this paper have been already published in Russian, see [6], [13]. However, due to the termination of Samara State University in 2015, the website of the electronic journal [13] is now unavailable, and in [6], we published only a little part (i.e. the material of Section III of this paper). Besides, this paper is the only English presentation of this subject, and we are going to use the objects introduced here in further publications.

II. Preliminaries

The main designations used in this paper were described in detail in [4], [6], [7]; let us repeat very briefly the most basic and important of them.

A nondeterministic finite automaton is a 5-tuple

$$K = (\, Q, \Sigma, \delta, S, F \,), \tag{1}$$

where:

- Q is the set of states of K;
- Σ is the considered alphabet;
- $\delta : Q \times \Sigma \to \mathcal{P}(Q)$ is the transition function ($\mathcal{P}$ is the set of subsets); we shall consider automata without ε-transitions only;
- $S, F \subseteq Q$ are the sets of initial and final states respectively.

Sometimes, to denote the transition from state q_1 to state q_2 in accordance with the transition function δ, we shall use notation

$$q_1 \xrightarrow[\delta]{a} q_2 .$$

And like [6] (and unlike [4]), we shall *not* denote the language of an automaton K by L; the details can be read in [6], however, this paper can be read without these details.

We shall denote the canonical automata for languages L and L^R (the last one is the mirror language for L) by

$$\widetilde{L} = (Q_\pi, \Sigma, \delta_\pi, \{s_\pi\}, F_\pi)$$

and

$$\widetilde{L^R} = (Q_\rho, \Sigma, \delta_\rho, \{s_\rho\}, F_\rho).$$

Let us list the remaining notation defined for automaton (1) in [4], [6], [7] (see the details in these papers):

- the input and output languages of state $q \in Q$, i.e., notation $\mathcal{L}^{in}_K(q)$ and $\mathcal{L}^{out}_K(q)$;
- state-marking functions

  $$\varphi^{in}_K : Q \to \mathcal{P}(Q_\pi) \quad \text{and} \quad \varphi^{out}_K : Q \to \mathcal{P}(Q_\rho);$$

- binary relation $\# \subseteq Q_\pi \times Q_\rho$;
- the basis automaton

  $$\mathcal{BA}(L) = (\hat{Q}, \Sigma, \hat{\delta}, \hat{S}, \hat{F}\,); \tag{2}$$

- for its state $\hat{q} = {}^{A}_{X} \in \hat{Q}$, we shall write $\alpha(\hat{q}) = A$ and $\beta(\hat{q}) = X$.
- if for some two states $q \in Q$ (of automaton (1)) and $\hat{q} = {}^{A}_{X} \in \hat{Q}$ (of automaton (2)), we have

  $$\varphi^{in}_K(q) \ni \alpha(\hat{q}) \quad \text{and} \quad \varphi^{out}_K(q) \ni \beta(\hat{q})\,,$$

  then we shall write $[q \ni \hat{q}]$;
- the universal automaton $\mathcal{COM}(L)$, see [5], [14].

Note that in the works cited above, we also give the *algorithms* for constructing all these objects; certainly, the algorithms are based on the given regular language L only.

Some examples of the defined objects were given in [4], [7]. Let us consider one of examples once again.

Let us consider the language accepted either by automaton given on Fig. 1 ...

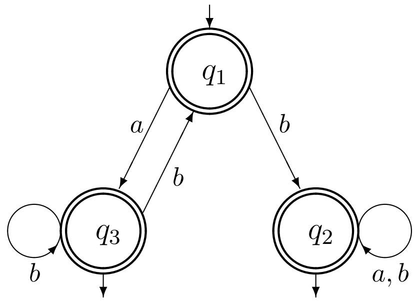

Fig. 1

... or by equivalent automaton *in canonical form* given on Fig. 2.

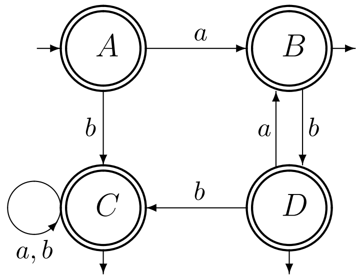

Fig. 2

And the following Fig. 3 and Table 1 (see also [4, Tab. 12]) show corresponding automaton $\widetilde{L^R}$ ...

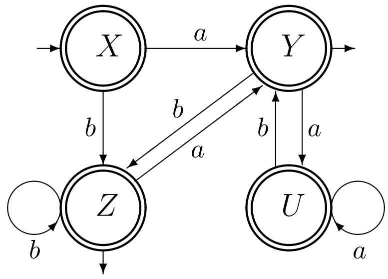

Fig. 3

... and binary relation $\#$ for the considered language:

Tab. 1

$\#$	X	Y	Z	U
A	$-$	$\#$	$\#$	$-$
B	$\#$	$-$	$\#$	$-$
C	$\#$	$\#$	$\#$	$\#$
D	$\#$	$\#$	$\#$	$-$

We also obtain the following automaton $\mathcal{BA}(L)$, see Tab. 2 below. Remark that it was not considered in [4], and was considered in [7]; we repeat the table of the last paper. Also, for the usability, we write

$${}^{A}_{X} \quad \text{instead of} \quad A\#X,$$

etc. Also we do *not* write symbols of sets (braces), i.e., we write

$$A\#X, A\#Y \quad \text{instead of} \quad \{\,{}^{A}_{X}, {}^{A}_{Y}\},$$

etc.

Tab. 2

	$\mathcal{BA}(L)$	a	b
$\to$	$A\#Y$	$B\#Z,\ B\#X$	$C\#U$
$\to$	$A\#Z$	–	$C\#Y,\ C\#Z,\ C\#X$
$\leftarrow$	$B\#X$	–	–
	$B\#Z$	–	$D\#Y,\ D\#Z,\ D\#X$
	$D\#Y$	$B\#Z,\ B\#X$	$C\#U$
$\leftarrow$	$C\#X$	–	–
	$C\#Y$	$C\#Z,\ C\#X$	$C\#U$
	$C\#Z$	–	$C\#Y,\ C\#Z,\ C\#X$
	$C\#U$	$C\#Y,\ C\#U$	–
$\leftarrow$	$D\#X$	–	–
	$D\#Z$	–	$C\#Y,\ C\#Z,\ C\#X$

III. Paths of the basis automaton and corresponding paths of an equivalent automaton

In this section and following ones, we consider a "natural" correspondence between loops of some automaton (1) defining the given regular language L and equivalent basis automaton $\mathcal{BA}(L)$. As we said before, a part of the material in this section was already published in Russian in [6].

For the considered language L and its basis automaton $\mathcal{BA}(L)$ (2), let us consider the transition function $\hat{\delta}$ of $\mathcal{BA}(L)$ as the *set* of its edges

$$\hat{\Delta} \;=\; \big\{\, \hat{e}_i = (\hat{p}_i, a_i, \hat{r}_i) \;\big|\; \hat{p}_i, \hat{r}_i \in \hat{Q},\ a_i \in \Sigma,\ \hat{\delta}(\hat{p}_i, a_i) \ni \hat{r}_i,\ i \in \{1, \dots, \hat{m}\} \big\}$$

(where $\hat{m}$ is the number of edges of automaton (2)). Thus, we have numbered ("colored") the edges of automaton $\mathcal{BA}(L)$.

And the transition function of automaton (1) can also be considered as the set of edges

$$\Delta \;=\; \big\{\, e_j{=}(p_j, a_j, r_j) \;\big|\; p_j, r_j \in Q,\ a_j{\in}\Sigma,\ \delta(p_j, a_j){\ni}r_j,\ j{\in}\{1, \dots, m\} \big\}$$

(where m is the number of edges of automaton (1)).

Let us mark each edge of automaton (1) *by a set* of possible (already used) "colors", i.e., let us consider function

$$\Omega : \Delta \to \mathcal{P}(\hat{\Delta})$$

constructed in the following way. For the edge

$$e_j = (p_j, a_j, r_j),$$

we set $\Omega(e_j) \ni \hat{e}_i$ (where $i \in \{1, \dots, \hat{m}\}$) if and only if the following conditions hold:

$$[p_j \ni \hat{p}_i]\,; \quad a_j = a_i\,; \quad [r_j \ni \hat{r}_i]\,.$$

Let us consider the set of paths of automaton $\mathcal{BA}(L)$; it is important to remark, that we consider *not only simple paths*. We can denote a path of automaton (2) by $\hat{\nu}$ and consider it as the sequence

$$\hat{\nu} = (\hat{e}^{\hat{\nu}}_1, \dots, \hat{e}^{\hat{\nu}}_{\hat{n}}), \tag{3}$$

where $\hat{n}$ is the number of its edges. Its k-th edge $\hat{e}_k$ will be denoted by

$$\hat{e}^{\hat{\nu}}_k = (\hat{p}^{\hat{\nu}}_k, a^{\hat{\nu}}_k, \hat{r}^{\hat{\nu}}_k)$$

($k \in \{1, \dots, \hat{n}\}$). Certainly, $\hat{p}^{\hat{\nu}}_{k+1} = \hat{r}^{\hat{\nu}}_k$ for each possible k.

For the path (3), let us define the set of *$\hat{\nu}$-corresponding paths* of automaton K. Thus, consider a path

$$\nu = (e^{\nu}_1, \dots, e^{\nu}_{\hat{n}}) \tag{4}$$

of automaton K. The value $\hat{n}$ is the same as in (3) (remember that both K and $\mathcal{BA}(L)$ do not contain ε-edges), and its k-th edge is

$$e^{\nu}_k = (p^{\nu}_k, a^{\nu}_k, r^{\nu}_k)$$

for each possible k. Let us say, that this path ***corresponds to*** $\hat{\nu}$, if for each $k \in \{1, \dots, \hat{n}\}$,

$$\Omega(e^{\nu}_k) \ni \hat{e}^{\hat{\nu}}_k\,.$$

(I.e., if each edge can be "colored by the color" of the edge of path $\hat{\nu}$ with the same number.) We shall also say, that for such paths, the states $\hat{p}^{\hat{\nu}}_k$ and p^{ν}_k are ***corresponding*** for each $k \in \{1, \dots, \hat{n}\}$.

Let us consider some examples. Consider the language which can be defined by regular expression

$$(a + ab + ba)^*. \tag{5}$$

Let us now consider the graphical description of the following automata.

- $\widetilde{L}$, see Fig. 4 below.
  Remark that in this example, we have $L = L^R$. Therefore considering Fig. 4 and changing marks A, B and C for X, Y and Z respectively, we obtain automaton $\widetilde{L^R}$.

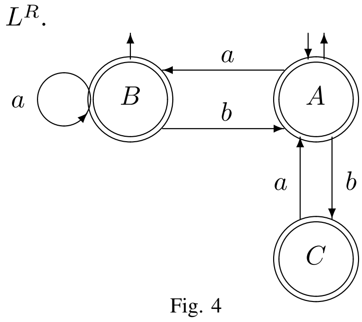

Fig. 4

- $\mathcal{BA}(L)$, see Fig. 5:

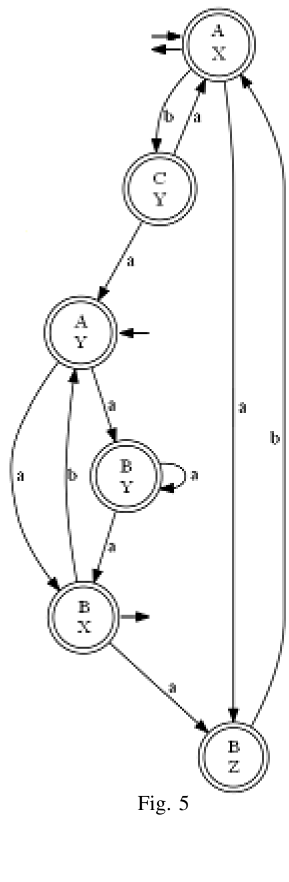

Fig. 5

- The equivalent automaton K, see Fig. 6 below.
  Like the agreements of [7], we write here also the values of state-marking functions.

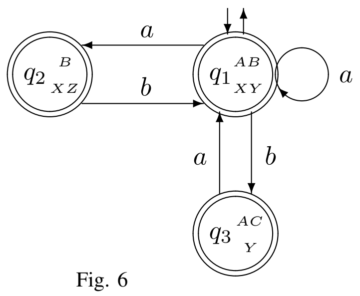

Fig. 6

For example, let us consider only two the following edges of automaton $\mathcal{BA}(L)$: the "green" edge

$$\substack{A\\Y} \xrightarrow[\hat{\delta}]{a} \substack{B\\X}$$

and the "blue" edge

$$\substack{B\\Y} \xrightarrow[\hat{\delta}]{a} \substack{B\\X}\,.$$

Therefore (Fig. 6), the following edges of automaton K (and *only they*) can be "green":

$$q_1 \xrightarrow[\delta_\pi]{a} q_1\,,\quad q_1 \xrightarrow[\delta_\pi]{a} q_2 \quad\text{and}\quad q_3 \xrightarrow[\delta_\pi]{a} q_1\,,$$

and the following edges (and *only they*) can be "blue":

$$q_1 \xrightarrow[\delta_\pi]{a} q_1 \quad\text{and}\quad q_1 \xrightarrow[\delta_\pi]{a} q_2\,.$$

Therefore we obtain, e.g., that the edge

$$q_1 \xrightarrow[\delta_\pi]{a} q_1$$

can be "colored" *in both* "green" and "blue" *colors* (and, maybe, in some other ones).

Let us also remark, that the set of simple loops of automaton $\mathcal{BA}(L)$ includes the loop

$$\substack{A\\Y} \xrightarrow[\hat{\delta}]{a} \substack{B\\Y} \xrightarrow[\hat{\delta}]{a} \substack{B\\X} \xrightarrow[\hat{\delta}]{b} \substack{A\\Y}\,. \tag{6}$$

We shall use this loop in some next examples.

IV. The first property of an automaton

In the example considered before, for each state $\substack{A\\X}$ of $\mathcal{BA}(L)$ there was a corresponding state of K containing all the corresponding loops. In other words, the following fact held in previous example.

Property 1: For each state $\substack{A\\X}$ of $\mathcal{BA}(L)$,
there exists a state q of K (where $[q \ni \substack{A\\X}]$),
such that for each loop ν of state $\substack{A\\X}$,
there exists a loop of K corresponding to ν. $\square$

The next example shows, that there exist some automata where Property 1 does not hold.[1] Thus, let us consider such example briefly, the considered automaton is given on Fig. 7:

[1] In the next sections, also in Part II, we shall consider some other examples for this thing. Certainly, for each such example, K has to contain a *path* corresponding to ν.

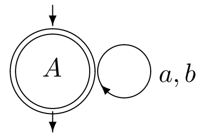

Fig. 7

Remark that we can think here, that automaton on Fig. 7 is simultaneously:

- $\widetilde{L}$;
- $(\widetilde{L^R})^R$ (changing mark A for X);
- $\mathcal{BA}(L)$ (changing mark A for $\substack{A\\X}$).

This automaton defines the whole language Σ^* (we can think that $\Sigma = \{a, b\}$). And both the states of equivalent automaton K on Fig. 8 ...

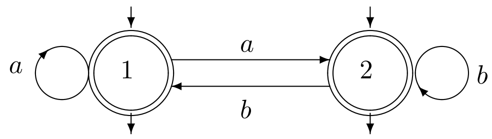

Fig. 8

... have the same values of φ_K^{in} and φ_K^{out} (i.e., $\{A\}$ and $\{B\}$ respectively), but neither of states of K has both loops corresponding to the loops of $\mathcal{BA}(L)$ with labels a and b.

V. Long corresponding loops

Let us define ***long loops***. Firstly, let a path (3) defined in Section III *be the loop*. Then let us define its *n-loop* (i.e., long loop) for each $n \geq 1$ in the usual way.

Definition 1: For $n \geq 1$, the ***n-loop*** of the loop $\hat{\nu} = (\hat{e}_1^{\hat{\nu}}, \dots, \hat{e}_{\hat{n}}^{\hat{\nu}})$ is defined by

$$\hat{\nu}^n = (\hat{e}_1^{\hat{\nu}^n}, \dots, \hat{e}_{n\cdot\hat{n}}^{\hat{\nu}^n}), \tag{7}$$

where for each possible i (i.e., $i \in \{1, \dots, (n-1)\cdot\hat{\nu}\}$) we think that

$$\hat{e}_i^{\hat{\nu}^n} = \hat{e}_{i+n}^{\hat{\nu}^n}. \qquad \square$$

For this definition, let us remark the following things. Firstly,

$$\hat{r}_{n\cdot\hat{n}}^{\hat{\nu}^n} = \hat{p}_1^{\hat{\nu}^n}\,.$$

Secondly, we used notation $\hat{e}$, $\hat{\nu}$, etc; but we can assume, that we defined such long loops not only for basis automaton, but also for each automaton (for example, for the given automaton (1)).

For such long loops, we consider corresponding paths of automaton (1) as we did in Section III. We shall call such path

$$\nu = (e_1^\nu, \dots, e_{n\cdot\hat{n}}^\nu), \tag{8}$$

by "long n-path" of automaton (1) corresponding to loop (3) of automaton (2).

By [15], path (8) is a loop if $r_{n\cdot\hat{n}}^\nu = p_1^\nu$; then we shall call it by (long) n-loop. (In fact, we have already considered some examples of such long loops in previous section.)

Let us consider once again regular language defined by (5) and its automaton $\mathcal{BA}(L)$ on Fig. 5. We describe the possible method of duplicating the loop. Let us remark in advance, that *we can simply consider Fig. 11 only* (i.e., consider it as *the given* automaton), proving the equivalence of that automaton and automaton on Fig. 9.

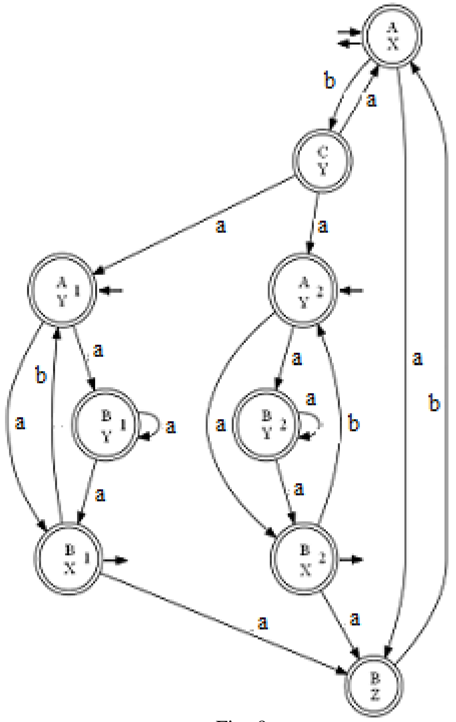

Fig. 9

(Remark also, that on the three following figures considered in this section, the right numbers mean indices: e.g., the notation

$$\genfrac{}{}{0pt}{}{A}{Y}\,1$$

marking a state means the mark

$$\binom{A}{Y}_1.$$

Certainly, the values of state-marking functions of *such* states are $\{A\}$ and $\{Y\}$; such facts also can be simply proved.)

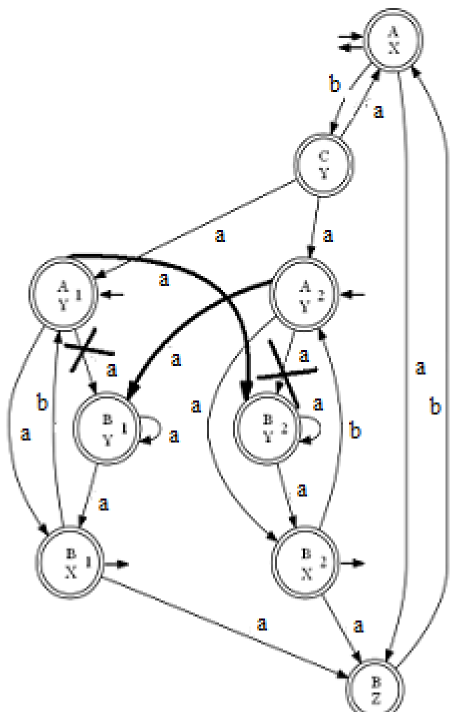

Fig. 10

Thus, let us add the duplicate of the loop (6); we obtain automaton on Fig. 9. Then using a simple operation (see Fig. 10, heavy and deleted lines there), we change:

- edge $\binom{A}{Y}_1 \xrightarrow{a} \binom{B}{Y}_1$ for edge $\binom{A}{Y}_1 \xrightarrow{a} \binom{B}{Y}_2$ (remark that we cannot use the notation $\hat{\delta}$ for transition function here, therefore we simply omit the name of this function);
- edge $\binom{A}{Y}_2 \xrightarrow{a} \binom{B}{Y}_2$ for edge $\binom{A}{Y}_2 \xrightarrow{a} \binom{B}{Y}_1$,

and obtain automaton on Fig. 11. (We think, that there is no need to describe such operation detailed.)

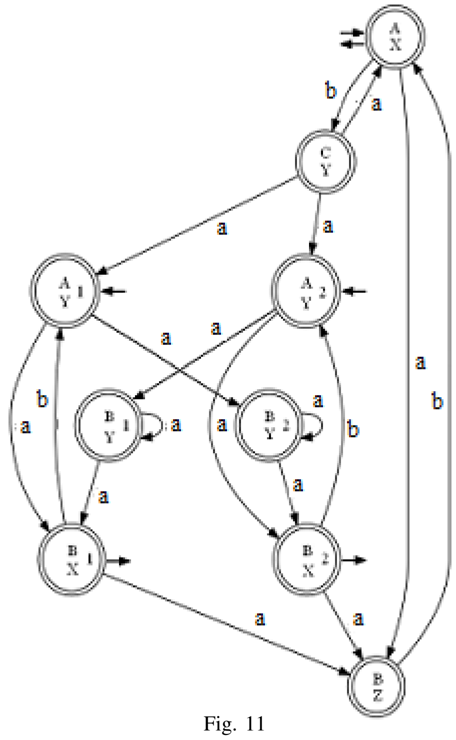

Fig. 11

The last automaton has, e.g., the long loop

$$\binom{A}{Y}_1 \xrightarrow{a} \binom{B}{Y}_2 \xrightarrow{a} \binom{B}{X}_2 \xrightarrow{b} \binom{A}{Y}_2$$
$$\xrightarrow{a} \binom{B}{Y}_1 \xrightarrow{a} \binom{B}{X}_1 \xrightarrow{b} \binom{A}{Y}_1$$

corresponding to the long loop

$$\genfrac{}{}{0pt}{}{A}{Y} \xrightarrow[\hat{\delta}]{a} \genfrac{}{}{0pt}{}{B}{Y} \xrightarrow[\hat{\delta}]{a} \genfrac{}{}{0pt}{}{B}{X} \xrightarrow[\hat{\delta}]{a} \genfrac{}{}{0pt}{}{A}{Y} \xrightarrow[\hat{\delta}]{a} \genfrac{}{}{0pt}{}{B}{Y} \xrightarrow[\hat{\delta}]{a} \genfrac{}{}{0pt}{}{B}{X} \xrightarrow[\hat{\delta}]{a} \genfrac{}{}{0pt}{}{A}{Y}$$

of the given basis automaton $\mathcal{BA}(L)$.

We shall continue to consider the last automaton in Part II of this paper.

VI. The second property of an automaton

Let us complicate the condition formulated in Section III. Thus, in the examples considered before (automata on *all* the figures considered before, *excepting Fig. 8*), the following condition held: for each state $\genfrac{}{}{0pt}{}{A}{X}$ of $\mathcal{BA}(L)$, there was a corresponding state of K containing all the *corresponding long* loops. (Recall that before the similar condition, i.e. Property 1, for corresponding *loops* only, not for long loops.) In other words, the following fact held in all previous examples of Section V.

Property 2: For each state $\genfrac{}{}{0pt}{}{A}{X}$ of $\mathcal{BA}(L)$, there exists a state q of K (where $[q \ni \genfrac{}{}{0pt}{}{A}{X}]$),

such that for each loop ν of state $\genfrac{}{}{0pt}{}{A}{X}$,
there exists a number $n \geq 1$,
such that there exists a q-loop corresponding to $\nu^{\,n}$. □

However. the considered before automaton on Fig. 8 shows, that there exist some automata where this fact *also* does not hold. (As before, K contains a *path* corresponding to $\nu^{\,n}$ for each $n \geq 1$.) And the following simplest Table 3 shows all the possibilities of fulfilling or not fulfilling the formulated properties for NFAs and corresponding examples of the automata considered before.

Tab. 3

Automaton	Property 1	Property 2
Fig. 7	$+$	$+$
Fig. 11	$-$	$+$
Fig. 8	$-$	$-$

In the following proposition, we consider the first of the properties of corresponding loops.

Proposition 1: For each loop of automaton $\mathcal{BA}(L)$ there exists $n \geq 1$, and for it there exists a corresponding n-loop of automaton K.

(Notice that we do *not* require that the given loop of automaton $\mathcal{BA}(L)$ is simple.)

Proof. Consider a loop

$$\hat{\nu} = \left(\hat{e}_1^{\hat{\nu}}, \ldots, \hat{e}_{\hat{n}}^{\hat{\nu}}\right)$$

of automaton $\mathcal{BA}(L)$, starting and finishing in state

$$\hat{p}_1^{\hat{\nu}^n} = \hat{r}_{\hat{n}}^{\hat{\nu}^n}.$$

Let v be its label ($v \neq \varepsilon$, because $\mathcal{BA}(L)$ has no ε-edges), let also

$$u \in \mathcal{L}^{in}_{\mathcal{BA}(L)}\left(\hat{p}_1^{\hat{\nu}^n}\right) \quad \text{and} \quad w \in \mathcal{L}^{out}_{\mathcal{BA}(L)}\left(\hat{r}_{\hat{n}}^{\hat{\nu}^n}\right);$$

remark that $uv^iw \in L$ for each $i \geq 0$.

Let n be greater than the number of states of automaton K. Consider word

$$x = uv^nw \in L;$$

because automaton $\mathcal{BA}(L)$ is unambiguous, it accepts x passing (n times) loop $\hat{\nu}$. Automaton K is equivalent (i.e., it defines L), then it accepts x.

Consider one of possible paths for automaton K defining x; by definition, some its sub-paths correspond to considered loop $\hat{\nu}$ of automaton $\mathcal{BA}(L)$. For this path of K, consider the set of states passed before reading the first letter of v in the word x; we have n such cases of reading v. Because the number of states of automaton K is less than n, we obtain the coincidence for at least 2 states of this set, and, therefore, the existence of corresponding long loop. □

We shall continue to consider long corresponding loops in Part II of this paper. Besides, we shall consider there some other properties of the states and loops of an arbitrary finite automaton, completing our approach to their classification. We also shall define so-called including loops, partially complete loops and complete cyclic states, consider some their properties and formulate an important hypothesis about them; in Introduction, we already gave brief information on these objects.

Remark in the conclusion of Part I, that in [6] we also considered cases, when a NFA defines not the given language (i.e., language of $\mathcal{BA}(L)$ according to the terminology used here), but some its *own subset*; in both parts of this paper, we shall not consider these issues.

REFERENCES

[1] Melnikov B. and Vakhitova A. *Some more on the finite automata*. The Korean Journal of Computational and Applied Mathematics (Journal of Applied Mathematics and Computing). 1998, vol. 5, no. 3. pp. 495–506.

[2] Melnikov B. and Melnikova A. *Edge-minimization of non-deterministic finite automata*. The Korean Journal of Computational and Applied Mathematics (Journal of Applied Mathematics and Computing). 2001, vol. 8, no. 3. pp. 469–479.

[3] Melnikov B. and Sayfullina M. *O nekotoryh algoritmah ekvivalentnogo preobrazovaniya nedeterminirovannyh konechnyh avtomatov [On some algorithms for the equivalent transformation of nondeterministic finite automata]*. Izvestiya of Higher Educational Institutions. Mathematics. 2009, no. 4, pp. 67–72. (in Russian, https://elibrary.ru/item.asp?id=11749888)

[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] Dolgov V. and Melnikov B. *Postroenie universal'nogo konechnogo avtomata … [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)

[6] Dolgov V., Melnikov B. and Melnikova A. *Cikly grafa perehodov bazisnogo avtomata … [Loops of the transition graph of a basic automaton and related questions]*. Vestnik of Voronezh State University. 2016, no. 4, pp. 95–111. (in Russian, https://elibrary.ru/item.asp?id=27257800)

[7] Melnikov B. *The complete finite automaton*. International Journal of Open Information Technologies. 2017, vol. 5, no. 10. pp. 9–17.

[8] Melnikov B. *A new algorithm of the state-minimization for the nondeterministic finite automata*. The Korean Journal of Computational and Applied Mathematics (Journal of Applied Mathematics and Computing). 1999, vol. 6, no. 2. pp. 277–290.

[9] Geldenhuys J., van der Merwe A. and van Zijl L. *Reducing Nondeterministic Finite Automata with SAT Solvers*. Proceedings of 8th International Workshop on Finite State Methods and Natural Language Processing, Lecture Notes in Artificial Intelligence. July 2009, vol. 6062, pp. 81–92.

[10] Kirsten D. *Distance desert automata and the star height problem*. Informatique Théorique et Applications. 2005, vol. 39, no. 3. pp. 455–509.

[11] 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.

[12] Bender E. *Mathematical Methods in Artificial Intelligence*. Wiley-IEEE Computer Society Press (Los Alamitos), 1996, 656 p.

[13] Melnikov B. *O zvyozdnoy vysote regulyarnogo yazyka … [On the star-height of a regular language. Part II: Auxiliary constructions]*. Heuristic algorithms and distributed computations. 2014, no. 2, pp. 69–81. (in Russian, https://elibrary.ru/item.asp?id=22030666)

[14] 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.

[15] Harary F. *Graph Theory*. Addison Wesley (Boston), 1969, 274 p.

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.