ISSN 0005-1179 (print), ISSN 1608-3032 (online), Automation and Remote Control, 2025, Vol. 86, No. 1, pp. 86–98.

OPTIMIZATION, SYSTEM ANALYSIS, AND OPERATIONS RESEARCH

Algebraic Methods of the Synthesis of Models Based on the Graphical Representation of Finite State Machines

V. V. Menshikh[*,a] **and V. A. Nikitenko**[*,b]

[*] *Voronezh Institute of the Ministry of Internal Affairs of Russia, Voronezh, Russia*
e-mail: [a]*menshikh@list.ru,* [b]*vitalijnikitenko82043@gmail.com*

Received April 10, 2024
Revised July 24, 2024
Accepted September 2, 2024

Abstract—The issues of modeling objects and systems based on the graphical representation of finite state machines using algebraic methods are considered. The problem of finite state machines synthesis based on the construction of the algebra of their graphoids is solved. With this aim existing operations on finite state machines are transferred to their graphoids. Subject to additional requirements, that may emerge during the analysis of the subject area new operations are introduced. This defines the algebra of finite state machines graphoids, which enables the synthesis of graphoids for finite state machines models using the algorithm proposed by the authors. Statements confirming the correctness of the algorithm are proven. A numerical example of the finite state machines model graphoid synthesis for the joint actions of functional groups in an emergency area.

Keywords: finite state machines graphoids, operations on graphoids, algebra of graphoids, the parallel synchronous change of automata states, parallel synchronous state transitions of automata of finite state machines states, invalid states, invalid vertex, graphoid synthesis

DOI: 10.31857/S0005117925010075

1. INTRODUCTION

An effective tool for modeling the dynamics of object and system functioning in various subject areas is finite automata [1]. However, as the complexity of the modeled objects and systems increases, the size of the input, output, and state alphabets of the automaton grows significantly. This substantially complicates the modeling process, makes the models excessively cumbersome, and hinders the interpretation of the modeling results. In this case, an effective approach is the use of a systemic methodology, according to which the object or system is initially decomposed into components, automata models for individual components are developed, and a general model is synthesized [2, 3]. The implementation of each of the aforementioned stages largely depends on the specific characteristics of the subject area of the modeled objects or systems. The most challenging stage is the synthesis of the general model, as it has the greatest impact on its adequacy. For instance, in [4, 5], automata theory methods were used to model the behavior of digital production twins based on automata algebra, which included not only well-known operations but also operations introduced by the authors to account for the peculiarities of the modeled object. Other examples of introducing domain-specific operations on automata and utilizing automata algebra can be found in [6–8]. Another challenge arising during the synthesis of automata models is the presence of significant constraints on the selection of possible components for the general model. If the component models must correspond to predefined objects or systems, specific requirements may be imposed on the synthesis process of the general model. These requirements include the necessity to eliminate invalid combinations of states in the components of the general model. An example of this is the task of modeling the joint actions of several functional groups involved in responding to an emergency situation [9, 10]. Constraints may arise, for instance, from the need to prevent conflicts between functional groups or to account for synergistic effects during their coordinated actions [11]. Automata that model the functioning of these groups serve as components of the general model for the emergency response process. It should be noted that the set of states of the components of the general model is often fully known. In this case, the automaton is represented as a labeled graph (a graphoid), where the vertices correspond to states, the edges represent transitions between states, and the edge weights describe the automaton's responses to various input symbols. This circumstance enables the synthesis of the graphoid of the general model by combining the graphoids of the automata models of its components. The solution to this problem can be achieved using algebraic methods. In [12, 13], the concept of an algebra of finite deterministic automata was introduced based on a set of composition operations, which are also applicable to automata graphoids. In particular, necessary and sufficient conditions for the decomposition of an automaton into a network of component automata were established using the introduced operations and by solving automaton equations through a specially defined language of paired algebras. The approaches used are naturally applicable to automata graphoids, whose representation is simply augmented with descriptions of the corresponding automata. However, in some subject areas, the solution obtained using the approach described in these works may fail to produce meaningful results because it does not account for potential constraints on the joint functioning of the components. In this regard, the task of developing a universal approach to synthesizing the graphoid of the general model, which takes into account the constraints on the joint functioning of the modeled objects or systems, is particularly relevant. In this work, this problem is addressed through the use of algebraic methods, and the correctness of the proposed approach is also substantiated.

2. THE ALGEBRA OF FINITE STATE MACHINES GRAPHOIDS

By the algebra $\mathcal{A} = \langle \mathcal{N}, \mathcal{S} \rangle$, according to [14], we mean the set $\mathcal{N}$ along with the operations defined on it.

$$\mathcal{S} = \{f_{11}, f_{12}, \dots, f_{1n_1}, f_{21}, f_{22}, \dots, f_{2n_2}, \dots, f_{m1}, f_{m2}, \dots, f_{mn_m}\},$$

where $\mathcal{N}$ is the carrier, and $\mathcal{S}$ is the signature of the algebra (f_{kl} is the lth k-ary operation).

A graphoid G of a finite deterministic non-initial abstract Moore automaton A is a quadruple [15] (Q, F, X, Y), where Q is the set of numbered vertices corresponding to the states of automaton A; F is the operator describing weighted edges, i.e., transitions between states and their corresponding output symbols depending on the input symbols; X is the input alphabet of automaton A; Y is the output alphabet of automaton A.

The notation $A \langle G$ will be used when automaton A corresponds to graphoid G.

We will now provide descriptions of operator F that are convenient for further use.

Expression

$$F^{x/y} q^i = q^{i_s}(x/y).$$

It means that if the automaton is in a state corresponding to the vertex of the graphoid q^i and the input symbol $x \in X$, is received, the automaton will transition to the state corresponding to the vertex of the graphoid q^{i_s} and an output symbol $y \in Y$ will be generated.

Denote

$$Fq^i = \bigcup_{\substack{x \in X \\ y \in Y}} \left\{ F^{x/y} q^i \right\}.$$

In these notations, the result of the operation of the operator F can be described as

$$\left\{Fq^i = \{q^{i_1}(x^{j_1}/y^{k_1}), \ldots, q^{i_l}(x^{j_l}/y^{k_l}), \ldots, q^{i_{n_i}}(x^{j_{n_i}}/y^{k_{n_i}})\}, i = \overline{1, |Q|}\right\}.$$

It is assumed that in general, $\{x^{j_1}, \ldots, x^{j_{n_i}}\} \subseteq X$, i.e., the automaton can be partial.

The operator F can be represented as a symbolic matrix, where the elements are pairs x/y. The algebra of such matrices described in [3] simplifies the process of developing numerical methods for operating with graphoids. However, this requires justification for the correctness of the operations used.

The introduction of the concept of the algebra of automata graphoids, where the carrier $\mathcal{N}$ is some set of graphoids $\mathcal{G}_0$, allows formalizing the procedure for synthesizing the graphoid of the general model of objects or systems, whose automata models are described by graphoids contained in the set $\mathcal{G}_0$, using various operations.

Let us now turn to the description and justification of the correctness of these operations.

3. ALGEBRA OF FINITE STATE MACHINE'S GRAPHOIDS OPERATIONS

Define the operation $\times$ on finite non-empty pairwise disjoint sets $M_1 = \left\{m_1^1, \ldots, m_{|M_1|}^1\right\}, \ldots,$ $M_n = \left\{m_1^n, \ldots, m_{|M_n|}^n\right\}$:

$$M_1 \times \ldots \times M_n = \left\{\left\{m_{i_1}^1, \ldots, m_{i_n}^n\right\} | i_1 = \overline{1, |M_1|}, \ldots, i_n = \overline{1, |M_n|}\right\}.$$

In this case $\times$ is not the Cartesian product because a result doesn't depend on order of operation. It enables to provide the commutativity of operations on of finite state machine's graphoids.

Suppose $G_1, G_2 \in \mathcal{G}_0$ — graphoids

$$G_1 = (Q_{G_1}, F_{G_1}, X_{G_1}, Y_{G_1}); \tag{1}$$
$$G_2 = (Q_{G_2}, F_{G_2}, X_{G_2}, Y_{G_2}). \tag{2}$$

If graphoids (1) and (2) satisfy the conditions

$$Y_{G_1} \cap X_{G_2} = \varnothing; \tag{3}$$
$$Y_{G_2} \cap X_{G_1} = \varnothing, \tag{4}$$
$$\Pi = G_1 \times G_2 = (Q_\Pi, F_\Pi, X_\Pi, Y_\Pi),$$

then $q_\Pi \in Q_\Pi$ will be define like $q_\Pi = \{q_{G_1}, q_{G_2}\}$ and $Q_\Pi, F_\Pi, X_\Pi, Y_\Pi$ will be set by equation:

$$Q_\Pi = Q_{G_1} \times Q_{G_2};$$
$$F_\Pi q_\Pi = F_{G_1} q_{G_1} \times F_{G_2} q_{G_2};$$
$$X_\Pi = X_{G_1} \times X_{G_2};$$
$$Y_\Pi = Y_{G_1} \times Y_{G_2}.$$

The graphoid Π corresponds to the parallel operation of automata described by graphoids G_1 and G_2, with synchronized state transitions.

If graphoids (1) and (2) satisfy the condition

$$X_{G_1} \cap X_{G_2} = \varnothing, \tag{5}$$

then the + of graphoids (1) and (2) is defined as the graphoid

$$\Sigma = G_1 + G_2 = (Q_\Sigma, F_\Sigma, X_\Sigma, Y_\Sigma),$$

where $q_\Sigma \in Q_\Sigma$ is defined as $q_\Sigma = \{q_{G_1}, q_{G_2}\}$, and $Q_\Sigma, F_\Sigma, X_\Sigma, Y_\Sigma$ are determined by the following formulas

$$Q_\Sigma = Q_{G_1} \times Q_{G_2};$$
$$F_\Sigma q_\Sigma = (F_{G_1} q_{G_1} \times \{q_{G_2}\}) \cup (\{q_{G_1}\} \times F_{G_2} q_{G_2});$$
$$X_\Sigma = X_{G_1} \cup X_{G_2};$$
$$Y_\Sigma = Y_{G_1} \cup Y_{G_2}.$$

The graphoid Σ corresponds to the parallel operation of automata described by graphoids G_1 and G_2, with asynchronous state transitions.

Condition (5) eliminates the possibility of ambiguity during state transitions after performing the + operation, i.e., it prevents the emergence of a nondeterministic automaton.

It should be noted that from the definition of the $\times$ and + operations for graphoids (1) and (2), it follows that

$$Q_{G_1 \times G_2} = Q_{G_1 + G_2}. \tag{6}$$

Thus, the algebra $\mathcal{A}_1 = \langle \mathcal{G}_1, \mathcal{S}_1 \rangle$, where $\mathcal{S}_1 = \{\times, +\}$ is described and possesses the following properties:

$$(G_1 \times G_2) \times G_3 = G_1 \times (G_2 \times G_3);$$
$$G_1 \times G_2 = G_2 \times G_1;$$
$$(G_1 + G_2) + G_3 = G_1 + (G_2 + G_3);$$
$$G_1 + G_2 = G_2 + G_1.$$

Consequently, the algebra $\mathcal{A}_1$ is a commutative semigroup with respect to each operation in the signature $\mathcal{S}_1$.

4. THE COMPOSITION OF GRAPHOIDS

When simulating real systems, it is necessary to take into account the characteristics of the subject area, which lead to additional requirements for the synthesis of automatic models, which in turn imposes some restrictions on the operations on their graphs. The most common requirements are:

1) changing the state of one object or system can trigger a change in the state of another object or system;

2) an object or system obtained by performing operations may contain invalid states.

In this connection, there is a need to extend the $\mathcal{S}_1$ signal of algebra $\mathcal{A}_1$ by introducing operations that allow taking into account the described features.

The first feature is implemented by introducing the concept "state-trigger," which assumes that the transition of one automaton into this state initiates the transition of another automaton to a certain specified state depending on its current state, i.e. at least one of the conditions (3) or (4) is not met. Consider the process of operation of the automata, which are described by graphs (1) and (2), in this case.

If the vertex $q^i_{G_1}$ corresponds to the state-trigger of $A_1\langle G_1$, automaton, which initiates state change in $A_2\langle G_2$ automaton, then for convenience of further description we will designate this vertex as ${}^{T_{G_2}}q^i_{G_1}$.

Let the $A_1\langle G_1$ have a trigger state corresponding to the vertex of ${}^{T_{G_2}}q^i_{G_1}$, and in this finite state machine corresponds to the output character $y^k_{G_1}$. This symbol simultaneously serves as an input symbol for the automaton $A_2\langle G_2$ and initiates a state transition in it as follows: if $A_2\langle G_2$ was in a certain state corresponding to the vertex $q^s_{G_2}$, and the set of its input symbols that trigger a state

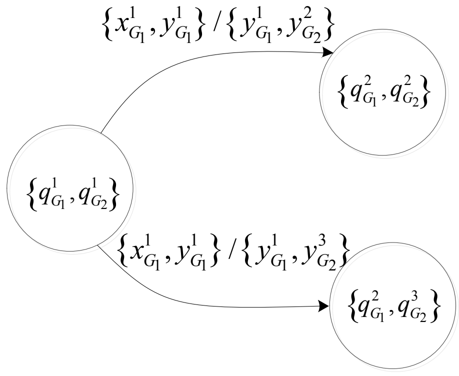

Fig. 1. Graphoid of the sub-automaton of automaton $A_{1,2}\langle K$.

transition includes $y^k_{G_1}$, then $A_2\langle G_2$ transitions from the state corresponding to the vertex $q^s_{G_2}$ a new state based on the input symbol $y^k_{G_1}$.

It is also possible that the automaton $A_2\langle G_2$ initiates state transitions in the automaton $A_1\langle G_1$, i.e., $A_2\langle G_2$ contains a trigger state corresponding to the vertex ${}^{T_{G_1}}q^i_{G_2}$.

Thus, there arises the need to describe a graphoid that represents the joint operation of two automatons, at least one of which contains a trigger state that influences the operation of the other.

If for graphoids (1) and (2) at least one of the conditions (3) or (4) is not satisfied, the composition $\circ$ of the graphoids is called a graphoid.

$$K = G_1 \circ G_2 = (Q_K, F_K, X_K, Y_K),$$

if Q_K, F_K, X_K, Y_K satisfy the following conditions:

$$Q_K = Q_{G_1} \times Q_{G_2};$$
$$F_K q_K = \bigcup_{\substack{t \in Y_{G_2} \\ l \in Y_{G_1}}} F^{t/l}_{G_1} q_{G_1} \times F^{l/t}_{G_2} q_{G_2};$$
$$X_K = X_{G_1} \times X_{G_2};$$
$$Y_K = Y_{G_1} \times Y_{G_2},$$

where $F^{t/l}_{G_1} q_{G_1}$is the transition mapping of the automaton from the state corresponding to the vertex q_{G_1} of graphoid G_1 when its input symbol $t \in X_{G_1} \cap Y_{G_2}$ appears, resulting in the output symbol $l \in Y_{G_1}$,

Similarly, $F^{l/t}_{G_2} q_{G_2}$ is the transition mapping of the automaton from the state corresponding to the vertex q_{G_2} of graphoid G_2, when its input symbol $l \in X_{G_2} \cap Y_{G_1}$ appears, resulting in the output symbol $t \in Y_{G_2}$.

It is worth noting that from the definition of the operations $\circ$, $\times$ and $+$ for graphoids (1) and (2), it follows that:

$$Q_{G_1 \circ G_2} = Q_{G_1 \times G_2} = Q_{G_1 + G_2}. \tag{7}$$

Proposition 1. *Let the graphoids* (1) *and* (2) *correspond to deterministic automata* A_1 *and* A_2, *and do not satisfy at least one of the conditions* (3) *or* (4). *Then,* $K = G_1 \circ G_2$ *is a graphoid of a deterministic automaton.*

Proof. Assume that the automaton $A_{1,2}\langle K$ is nondeterministic. Without loss of generality, we can assume that it contains a subautomaton whose graphoid is shown in Fig. 1, where $y^1_{G_1} = x^1_{G_2}$ appears.

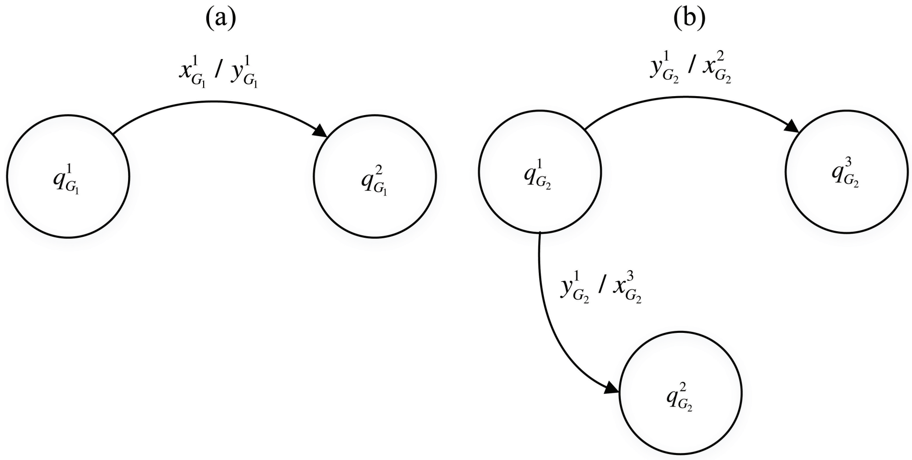

Fig. 2. (a) Graphoid of the sub-automaton A_1, (b) graphoid of the sub-automaton A_2.

Then, in automata A_1 and A_2, there will be subautomata whose graphoids are shown in Fig. 2.

Therefore, in this case, automaton A_2 would be nondeterministic, which contradicts the condition of the theorem.

Note that due to the associativity and commutativity of the operations $\times, \cup$, the composition operation $\circ$ also possesses these properties:

$$(G_1 \circ G_2) \circ G_3 = G_1 \circ (G_2 \circ G_3);$$
$$G_1 \circ G_2 = G_2 \circ G_1.$$

It is evident that if conditions (3) and (4) are satisfied, then $G_1 \circ G_2 = G_1 \times G_2$.

Let us define a new algebra $\mathcal{A}_2 = \langle \mathcal{G}_2, \mathcal{S}_2 \rangle$ with the signature $\mathcal{S}_2 = \{\circ, \times, +\}$, which, like the algebra $\mathcal{A}_1$, forms a commutative semigroup under each operation.

5. THE OPERATION OF GRAPHOID FILTERING

After performing the binary operations of the signature $\mathcal{S}_2$ a new graphoid of the algebra $\mathcal{A}_2$ is obtained. The automaton corresponding to this graphoid may not satisfy the constraints for the joint functioning of the components of the overall model because the resulting automaton may contain invalid combinations of their states, considering the specific characteristics of the domain. That is, conflict situations arise in the resulting automaton (it is assumed that the set of conflict situations is defined by the decision-maker). As a result, there is a need to introduce a filtering operation ∇, which allows excluding vertices of the graphoid that correspond to invalid states of the automaton. These vertices will also be referred to as invalid.

It should be noted that in [12, 13], only the issues of finding and removing unreachable states of automata were studied, which helps reduce the dimensionality of the problem, but may not meet the requirements of the domain.

Let $\Omega_1, \ldots, \Omega_n$ be the generators of the algebra $\mathcal{A}_2$, i.e., the graphoids from which, using the operations signature $\mathcal{S}_2$ of the all other graphoids of the carrier $\mathcal{G}_2$ can be obtained.

Let us define the set $\Psi = \{\Psi_1, \ldots, \Psi_k, \ldots, \Psi_r\}$ of invalid vertices of some graphoid. Each vertex Ψ_k corresponds to the set $\left\{q^{l_{k_1}}_{\Omega_{k_1}}, \ldots, q^{l_{|\Psi_k|}}_{\Omega_{k_{|\Psi_k|}}}\right\}$ of vertices, which are the generators of the algebra $\mathcal{A}_2$.

Let the graphoid $H = (Q_H, F_H, X_H, Y_H)$ be obtained by transforming the graphoid $G_1, \ldots, G_m \in \mathcal{G}_2$ using the operations of the signature $\mathcal{S}_2$. Then, the set of all vertices of the graphoid H is as follows:

$$Q_H = \Big\{Q_{G_1}, \ldots, Q_{G_m}, \{Q_{Gij} | Q_{Gij} = Q_{G_i} \times Q_{G_j}, \forall i, j \in \{1, m\}, i \neq j\}, \ldots,$$
$$\{Q_{G_{i_1,\ldots,i_n}} | Q_{G_{i_1,\ldots,i_n}} = Q_{G_{i_1}} \times \ldots \times Q_{G_{i_n}}, \forall i_1, \ldots, i_n \in \{1, m\}, i_1 \neq \ldots \neq i_n\}\Big\}.$$

Let us define the function

$$\pi(q_H, \Psi_k) = \begin{cases} 1, & \text{if } \Psi_k \subseteq q_H \\ 0, & \text{otherwise.} \end{cases}$$

Then the vertex $q_H \in Q_H$ is invalid if:

$$\sum_{k=1}^{r} \pi(q_H, \Psi_k) \neq 0.$$

Let us denote as

$$\Xi_H = \left\{ q_H \in Q_H | \sum_{k=1}^{r} \pi(q_H, \Psi_k) \neq 0 \right\}$$

a set of invalid vertexes.

For exclude invalid states, we introduce a unary filtering operation ∇.

The graphoid $\nabla_{\Xi_H} H$ is called the filtration of the graphoid H by the set Ξ_H if it is a subgraph of the graphoid H with the set of vertices $Q_{\nabla_{\Xi_H} H} = Q_H \setminus \Xi_H$.

Thus, the algebra $\mathcal{A}_3 = \langle \mathcal{G}_3, \mathcal{S}_3 \rangle$ with the signature $\mathcal{S}_3 = \{\nabla, \circ, \times, +\}$ is obtained.

6. ALGORITHM FOR GRAPHOID SYNTHESIS OF A FINITE STATE MACHINE MODEL

To develop this algorithm, we first define the algebra of graphoids, whose signature contains only the operations necessary to solve the problem of synthesizing graphoids for automaton models, taking into account the features described above.

Consider automata functioning simultaneously, which are described by graphoids (1) and (2). Their state transitions can occur either simultaneously or at different times. Therefore, the operation of the automata can be:

- Either synchronized, which in the synthesis process is described by:
  - — the operation $\times$, if they do not contain triggers that influence each other's functioning;
  - — the operation $\circ$, if they contain triggers that influence each other's functioning;
- or asynchronous, which is described by the operation $+$.

Both possibilities for the functioning of the automata must be taken into account during the synthesis process. Based on this, it is necessary to combine the operations $\circ, \times$ and $+$. For this, we introduce the union operation $\cup$ of graphoids.

Let the condition (5) hold for graphoids (1) and (2), and $Q_{G_1} = Q_{G_2}$, then the union $\cup$ of the graphoids is called the graphoid

$$C = G_1 \cup G_2 = (Q_C, F_C, X_C, Y_C),$$

where Q_C, F_C, X_C, Y_C are defined by following formulas:

$$Q_C = Q_{G_1} = Q_{G_2};$$
$$F_C q_C = F_{G_1} q_{G_1} \cup F_{G_2} q_{G_2};$$
$$X_C = X_{G_1} \cup X_{G_2};$$
$$Y_C = Y_{G_1} \cup Y_{G_2}.$$

This operation has the following properties:

$$(G_1 \cup G_2) \cup G_3 = G_1 \cup (G_2 \cup G_3)\,;$$
$$G_1 \cup G_2 = G_2 \cup G_1.$$

Considering the properties (6) and (7) of the operations $\circ$, $\times$ and $+$ introduced above, the synthesis process of graphoids for automaton models can be carried out using the following combinations of these operations:

$$G_1 \otimes G_2 = (G_1 \times G_2) \cup (G_1 + G_2);$$
$$G_1 \odot G_2 = (G_1 \circ G_2) \cup (G_1 + G_2).$$

The graphoids obtained after performing the operations $\otimes$ and $\odot$ may correspond to automata that contain unacceptable states. To exclude them, the filtration operation ∇ must be used.

The above leads to the conclusion that the algebra $\mathcal{A} = \langle \mathcal{G}, \mathcal{S} \rangle$, where $\mathcal{S} = \{\nabla, \otimes, \odot\}$, can be used for synthesizing graphoids of automaton models.

Now, let's consider the algebraic properties of the operations in the signature $\mathcal{S}$.

From the commutativity and associativity of the operations $\circ, \times, +, \cup$, it follows that the operations $\otimes$ and $\odot$ are also commutative and associative. Therefore, the order in which the synthesis of the overall model is carried out using these operations does not matter.

Let $\bullet$ denote one of the operations in the set $\{\otimes, \odot\}$. Suppose the graphoids $G_{i_1}, G_{i_2}, G_{i_3}, \ldots, G_{i_s}$ are obtained by transforming the graphoids $G_1, \ldots, G_m \in \mathcal{G}$ using operations from the signature $\mathcal{S}$. Then the following statement holds.

Proposition 2.

$$\nabla(G_{i_1} \bullet G_{i_2} \bullet G_{i_3} \bullet \ldots \bullet G_{i_s}) = \nabla\Big(\nabla\Big(\ldots\Big(\nabla((\nabla G_{i_1}) \bullet G_{i_2}) \bullet G_{i_3}\Big) \bullet \ldots\Big) \bullet G_{i_s}\Big).$$

Proof. We will use the method of mathematical induction.

Let $\Xi_{G_{i_t}}$ be the set of unacceptable vertices of the graphoid G_{i_t}, and let Ξ contain all possible unacceptable vertices of combinations of the graphoids $G_{i_1}, G_{i_2}, G_{i_3}, \ldots, G_{i_s}$.

(1) Base case $s = 2$. We need to prove that $\nabla(G_{i_1} \bullet G_{i_2}) = \nabla\big((\nabla G_{i_1}) \bullet G_{i_2}\big)$.
The set of vertices $\nabla(G_{i_1} \bullet G_{i_2})$ is $Q_{G_{i_1} \bullet G_{i_2}} = \widehat{Q}_{G_{i_1} \bullet G_{i_2}} \setminus \Xi$, and the set of vertices of $\nabla\big((\nabla G_{i_1}) \bullet G_{i_2}\big)$ is $Q_{G_{i_1} \bullet G_{i_2}} = ((\widehat{Q}_{G_{i_1}} \setminus \Xi_{G_{i_1}}) \times \widehat{Q}_{G_{i_2}}) \setminus \Xi$.
Now, let us transform the last expression:

$$Q_{G_{i_1} \bullet G_{i_2}} = \Big((\widehat{Q}_{G_{i_1}} \setminus \Xi_{G_{i_1}}) \times \widehat{Q}_{G_{i_2}}\Big) \setminus \Xi = \Big((\widehat{Q}_{G_{i_1}} \times \widehat{Q}_{G_{i_2}}) \setminus (\Xi_{G_{i_1}} \times \widehat{Q}_{G_{i_2}})\Big) \setminus \Xi. \tag{8}$$

In the set $\Xi_{G_{i_1}} \times \widehat{Q}_{G_{i_2}}$ all vertices are unacceptable, so, $\Xi_{G_{i_1}} \times \widehat{Q}_{G_{i_2}} \subseteq \Xi$, and we can rewrite the expression (6) as: $(\widehat{Q}_{G_{i_1}} \times \widehat{Q}_{G_{i_2}}) \setminus \Xi$, which corresponds to $\nabla(G_{i_1} \bullet G_{i_2})$.

(2) Assume that the statement is true for $s = k$ We now need to prove that it holds for $s = k+1$.
We have:

$$\nabla(G_{i_1} \bullet G_{i_2} \bullet G_{i_3} \bullet \ldots \bullet G_{i_s} \bullet G_{i_{s+1}})$$
$$= \nabla\Big(\nabla\Big(\nabla\Big(\ldots\Big(\nabla(\nabla(G_{i_1}) \bullet G_{i_2}) \bullet G_{i_3}\Big) \bullet \ldots\Big) \bullet G_{i_s}\Big) \bullet G_{i_{s+1}}\Big),$$

since the graphoids $G_{i_1}, G_{i_2}, G_{i_3}, \ldots, G_{i_s}$ are obtained by transformations using the operations of the signature $\mathcal{S}$, the expression $G_{i_1} \bullet G_{i_2} \bullet G_{i_3} \bullet \ldots \bullet G_{i_s}$ can be replaced by the equivalent graphoid $H = G_{i_1} \bullet G_{i_2} \bullet G_{i_3} \bullet \ldots \bullet G_{i_s}$. Thus, we obtain the expression $\nabla(G_{i_1} \bullet G_{i_2} \bullet G_{i_3} \bullet \ldots \bullet G_{i_s} \bullet G_{i_{s+1}})$ is $\nabla(H \bullet G_{i_{s+1}})$. Therefore, it is necessary to show the validity of the equality $\nabla(H \bullet G_{i_{s+1}}) = \nabla\big((\nabla H) \bullet G_{i_{s+1}}\big)$, which was proven in part 1).

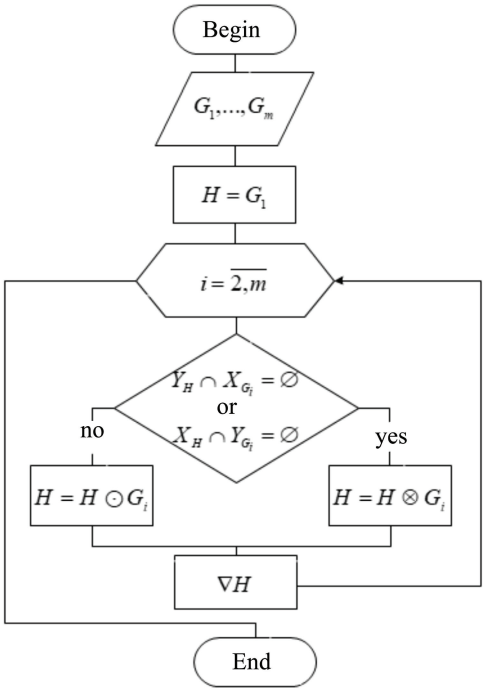

Fig. 3. Algorithm for synthesizing graphoids of automaton models.

Thus, the algorithm for synthesizing graphoids from $\mathcal{G}$ is as shown in Fig. 3. Its correctness follows from the algebraic properties of the operations of the signature $\mathcal{S}$ and the theorem proven above.

7. NUMERICAL EXAMPLE

As an example, consider the process of access control to an emergency zone, the organization of search and rescue operations, and the evacuation of people and material assets from this zone [10], which involves the use of the following standard functional groups:

(1) Organization of access to the emergency zone, whose actions are modeled by automaton $A_1\langle(G_1 = (Q_{G_1}, F_{G_1}, X_{G_1}, Y_{G_1}))$;

(2) Organization of the search for people and material assets to be evacuated, whose actions are modeled by automaton $A_2\langle(G_2 = (Q_{G_2}, F_{G_2}, X_{G_2}, Y_{G_2}))$;

(3) Organization of evacuation to a safe area, whose actions are modeled by automaton $A_3\langle(G_3 = (Q_{G_3}, F_{G_3}, X_{G_3}, Y_{G_3}))$.

During the development of the emergency situation, the listed functional groups can be in states corresponding to the vertices indicated in table.

Description of the vertices of the graphoids corresponding to the states of the automata $A_1\langle G_1, A_2\langle G_2, A_3\langle G_3$, modeling the actions of functional groups

$q^1_{G_1}$	Full perimeter control of the emergency zone
$q^2_{G_1}$	Implementation of access control
$q^1_{G_2}$	Waiting in the initial area
$q^2_{G_2}$	Movement to the search area
$q^3_{G_2}$	Searching for people and material valuables to be evacuated
$q^4_{G_2}$	Escorting people and material valuables to the assembly evacuation point
$q^1_{G_3}$	Waiting for the formation of the evacuation convoy
$q^2_{G_3}$	Accounting for the injured and forming the evacuation convoy
$q^3_{G_3}$	Movement to the safe zone

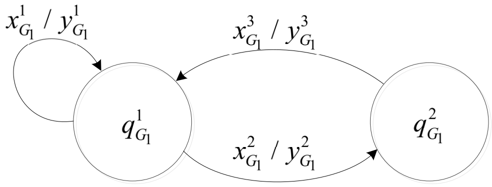

Fig. 4. Graphoid G_1.

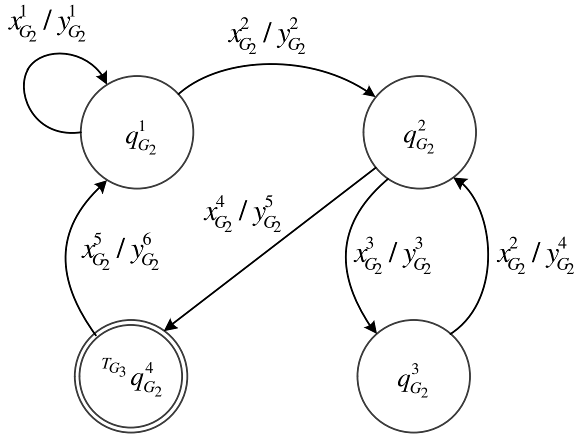

Fig. 5. Graphoid G_2.

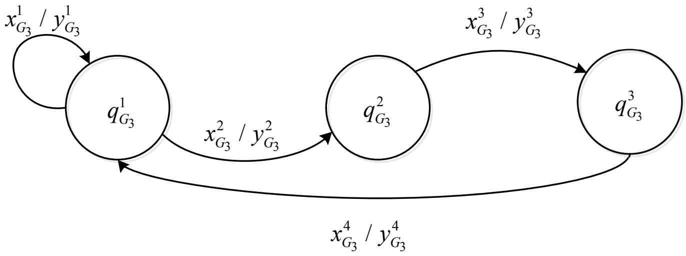

Fig. 6. Graphoid G_3.

The graphoids G_1, G_2, G_3 of finite state machines A_1, A_2, A_3 are shown in Figs. 4–6.

It is necessary to synthesize, using the developed algebra $\mathcal{A}$ a graphoid H, that defines the joint activities of the functional groups. The analysis of the task revealed:

1) potential conflict situations and the determination of the set of unacceptable vertices

$$\Xi = \left\{\left\{q^1_{G_1}, q^2_{G_2}\right\}, \left\{q^2_{G_1}, q^1_{G_2}\right\}, \left\{q^2_{G_1}, q^3_{G_2}\right\}, \left\{q^2_{G_1}, q^4_{G_2}\right\}, \left\{q^1_{G_1}, q^4_{G_2}, q^1_{G_3}\right\}, \left\{q^1_{G_1}, q^4_{G_2}, q^3_{G_3}\right\}\right\};$$

2) the need for automaton A_3 to be triggered by automaton A_2: if the output symbol of automaton A_2 is $y^5_{G_2}$, then the input symbol of automaton A_3 is $x^2_{G_3}$.

For example, the state corresponding to the vertex is a conflict state because the functional group described by automaton A_1, is advancing into the search area at the moment when the functional group described by automaton A_2, is performing territorial control.

We will describe the process of synthesizing the graphoid H in accordance with the algorithm shown in Fig. 1.

In the first iteration, the synthesis of graphoids H and G_2 is carried out:

It is assumed that $H = G_1$ and $Y_H = Y_{G_1}$;

Since there are no state triggers, as $Y_H \cap X_{G_2} = \varnothing$, the operation $H = H \otimes G_2$ is performed;

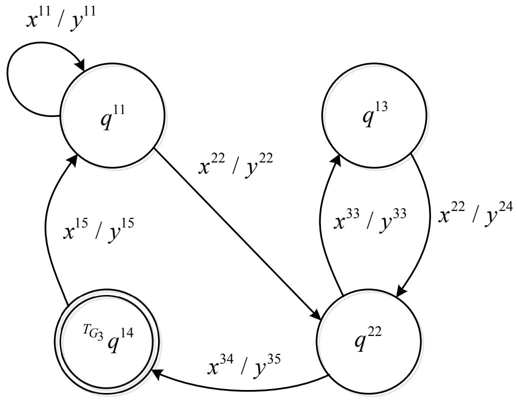

Fig. 7. Graphoid $H = \nabla(G_1 \otimes G_2)$.

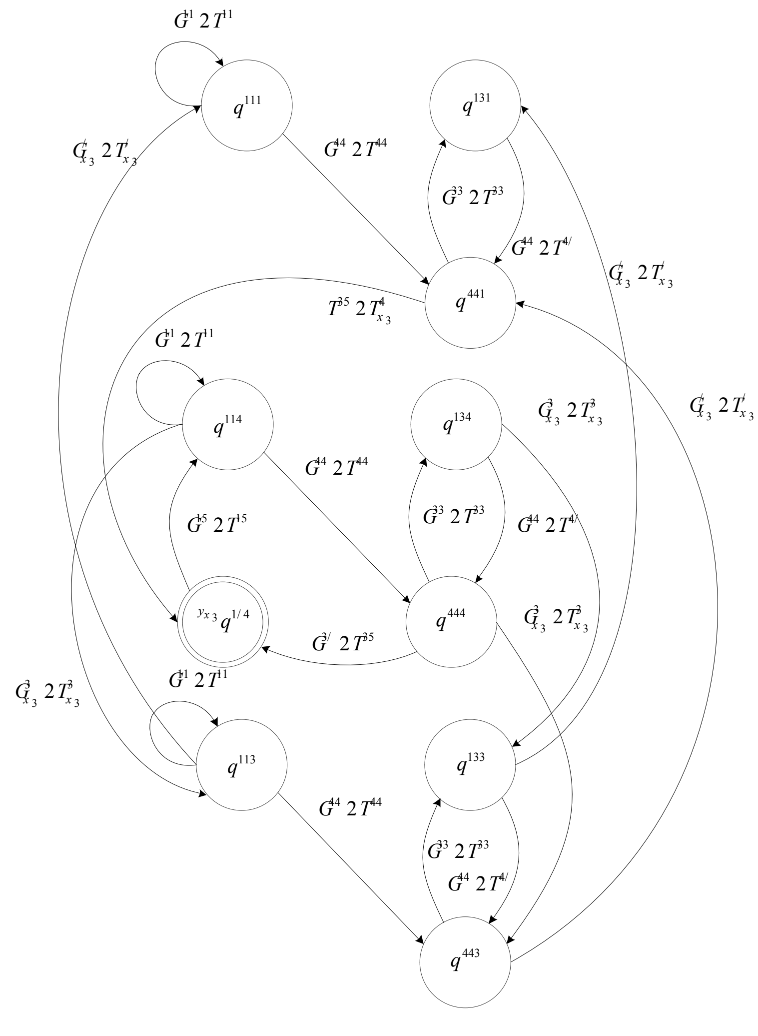

Fig. 8. Graphoid $= \nabla(\nabla(G_1 \otimes G_2) \odot G_3)$.

Invalid states, determined by the vertices $\{q^1_{G_1}, q^2_{G_2}\}$, $\{q^2_{G_1}, q^1_{G_2}\}$, $\{q^2_{G_1}, q^3_{G_2}\}$, $\{q^2_{G_1}, q^4_{G_2}\}$ are excluded from graphoid H, i.e., the operation $H = \nabla H$ is performed. The resulting graphoid is shown in Fig. 7, where the vertices $q^{ij} = \{q^i_{G_1}, q^j_{G_2}\}$, input symbols $x^{ij} = \{x^i_{G_1}, x^j_{G_2}\}$ and output symbols $y^{ij} = \{y^i_{G_1}, y^j_{G_2}\}$ are defined.

In the next iteration, the synthesis of graphoids H and G_3 is performed:

Since $Y_H \cap X_{G_3} \neq \varnothing$ the operation $H = H \odot G_3$ is performed;

Invalid vertices $\{q^1_{G_1}, q^4_{G_2}, q^1_{G_3}\}, \{q^1_{G_1}, q^4_{G_2}, q^3_{G_3}\}$ are excluded from graphoid H, i.e., the operation $H = \nabla H$, is performed. The resulting graphoid is shown in Fig. 8, with vertices $q^{ijk} = \{q^i_{G_1}, q^j_{G_2}, q^k_{G_3}\}$.

Thus, the graphoid H corresponds to an automaton that describes the parallel synchronous and asynchronous functioning of automata A_1, A_2, as well as the initialization of state transitions of automaton A_3.

In the resulting graphoid H, all interrelated actions of the three functional groups are considered. Their activities are directed at controlling access to the emergency zone, organizing the search for victims, and evacuating people and material valuables from the zone.

8. CONCLUSION

The article presents an algebra of graphoids of automata, which allows synthesizing a graphoid for the general model of automaton functioning. In constructing this algebra, operations on automata were partially transferred to the graphoids of these automata, and operations were introduced to account for additional domain-specific requirements. An algorithm for synthesizing the graphoids of automata based on this algebra has been developed, which allows constructing a generalized model of object functioning independently of the sequence in which they are connected, due to the commutativity of the operations. A numerical example of synthesizing the graphoid of an automaton is provided, describing the interrelated actions of three functional groups used in the event of an emergency. These functional groups carry out control of access to the emergency zone, organize the search for victims, and evacuate people and material valuables from the zone. As a result, a mathematical apparatus has been developed, enabling the modeling of joint actions of the functional groups involved in emergency response. This mathematical apparatus can later be used in models for assessing the effectiveness of functional group actions and optimizing the selection of their composition and tactics, by populating it with the contents of the input and output symbols of the automata.

REFERENCES

1. Kalman, R.E., Falb, P.L., and Arbib, M.A., *Topics in Mathematical System Theory*, New York: McGraw-Hill, 1969. Translated under the title *Ocherki po matematicheskoi teorii sistem*, Moscow: Mir, 1971.

2. Sysoev, V.V., Menshikh, V.V., Solodukha, R.A., and Zabiyako, S.V., Study of Interactions in a Network of Finite Deterministic Automata, *Radio Engineering*, 2000, no. 9, pp. 65–67.

3. Menshikh, V.V. and Nikitenko, V.A., Numerical Method for Aggregating Automaton Models Using Algebraic Operations on Automata, *Control Problems*, 2000, no. 9, pp. 65–67.

4. Gapanoich, D.A. and Sukhomlin, V.A., Algebra of Finite Automata as a Mathematical Model of Digital Twin for Smart Manufacturing, *Modern Information Technologies and IT Education*, 2022, vol. 18, no. 2, pp. 353–366.

5. Gapanoich, D.A. and Sukhomlin, V.A., Modeling of Mine Operations Using Algebra of Finite Automata DTA, *Modern Information Technologies and IT Education*, 2022, vol. 18, no. 3, pp. 634–643.

6. Volkova, K.M., Bulletin of the South Ural State University. Series: Computer Technologies, *Automatic Control, Radio Electronics*, 2022, vol. 21, no. 1, pp. 49–58.

7. Shirokova, E.V. and Yevtushenko, N.V., Synthesis of Safe Web Service Components Based on the Solution of Automaton Equations, *Modern Science: Actual Problems of Theory and Practice. Series: Natural and Technical Sciences*, 2023, no. 8, pp. 143–150.

8. Solovyev, V.V., Synthesis of Fast Finite Automata on Programmable Logic Integrated Circuits by Splitting Internal States, *Izv. RAS. Theory and Systems of Control*, 2022, no. 3, pp. 69–80.

9. Menshikh, V.V. and Korchagin, A.V., Structural Models of Interactions of Units of Law Enforcement Agencies in Case of Technogenic Emergencies, *Proceedings of the Academy of Management of the Ministry of Internal Affairs of Russia*, 2015, no. 2(34), pp. 54–58.

10. Menshikh, V.V., Samorokovskyi, A.F., Sereda, E.N., Gorlov, and V.V., *Modeling Collective Actions of Internal Affairs Officers*, Voronezh: Voronezh Institute of the Ministry of Internal Affairs of the Russian Federation, 2017.

11. Menshikh, V.V., Gorlov, V.V., and Nikitenko, V.A., Consideration of Synergistic Effects in the Composition of Automaton Models of Actions of Units of Law Enforcement Agencies Involved in Emergency Response, *Bulletin of Voronezh Institute of the Ministry of Internal Affairs of Russia*, 2023, no. 2, pp. 60-68.

12. Harmanis, J. and Stearns, R., *Algebraic Structure Theory of Sequential Machines*, New York: Prentice-Hall Inc., 1966.

13. Villa, T., Yevtushenko, N., Brayton, R.K., Mishchenko, A., Petrenko, A., and Sangiovanni Vincentelli, A.L., *The Unknown Component Problem: Theory and Applications*, Springer, 2012.

14. Gorbatov, V.A., *Fundamental'nye osnovy diskretnoi matematiki. Informatsionnaya matematika* (Fundamentals of Discrete Mathematics. Information Mathematics), Moscow: Nauka, 2000.

15. Melikhov, A.N., *Orientirovannye grafy i konechnye avtomaty* (Directed Graphs and Finite Automata), Moscow: Nauka, 1971.

This paper was recommended for publication by M.F. Karavai, a member of the Editorial Board