ISSN 0147-6882, Scientific and Technical Information Processing, 2023, Vol. 50, No. 6, pp. 602–609. © Allerton Press, Inc., 2023.
Russian Text © The Author(s), 2022, published in Iskusstvennyi Intellekt i Prinyatie Reshenii, 2022, No. 2, pp. 17–26.

System-Object Determinant Analysis. Partitive Classification Using a Formal-Semantic Normative System

S. I. Matorin[a, b, *] **and V. V. Mikhelev**[a]
[a] *National Research University Belgorod State University, Belgorod, Russia*
[b] *Belgorod University of Cooperation, Economics & Law, Belgorod, Russia*
**e-mail: matorin@softconnect.ru*
Received February 16, 2022

Abstract—This paper considers the construction of a partitive classification for conducting a system-object determinant analysis in a formal-semantic normative system. A formal semantic alphabet and rules for its use are described in terms of a descriptive logic. The algorithms developed provide computer support for the decomposition of a complex system in graphical-analytical modeling. An illustrative example is given.

Keywords: system-object determinant analysis, partitive classification, formal-semantic normative system, system-object approach Node–Function–Object, decomposition

DOI: 10.3103/S0147688223060114

INTRODUCTION

The need for the graphic-analytical modeling of complex objects and systems arises in many areas of human activity and, in particular, when there are not enough mathematical modeling tools. These include in reengineering business processes or designing organizational, technical or information systems [1–3]. An integral part of the graphic-analytical modeling of an object is the decomposition of more general model blocks into more specific ones, i.e., in the construction of a partitive classification of the elements of the modeled object [4, 5]. Until the present, this decomposition procedure is not only not automated but also does not have at least some kind of computer support. At one time, this circumstance even gave rise to a discussion of the following question on the internet: "What modeling tool that deprives the developer of that part of the 'creative' possibilities that lead to the diversity of the representation of organizational models will bring the least harm to the organization?" [6]. However, this discussion did not produce any results.

Decomposition (the building of a partitive classification) of a complex system is also seen in system-object determinant analysis (SODA), as proposed in [7]. SODA is an effective tool for analyzing the causes of a system, the stages of its formation and its existing properties. This knowledge is necessary for the analysis of existing systems, as well as for the creation of new systems in the process of their design. The construction of a partitive (integer-private) classification (or meronomy) of the analyzed or projected system, i.e., its decomposition, is performed in the last (third) stage of SODA. To formalize and algorithmize the decomposition procedure in partitive classifications within the framework of the system-object approach "Node–Function–Object," a formal-semantic normative system (FSNS) has been developed [8, 9].

An FSNS is a formal semantic alphabet of structural (nodal) elements and the rules for its use. The alphabet of nodes itself is based on a classification of types of connections/flows between nodes. In the mentioned works, the alphabet of nodal elements is described by means of the descriptive logic ALCHIO(D) [10, 11] by introducing a hierarchy of concepts of links/flows: *Material connection* (**m**) $\sqsubset$ *Connection* (**L**); *Information connection* (**i**) $\sqsubset$ *Connection* (**L**); *Real connection* (**v**) $\sqsubset$ *Material connection* (**m**); *Energy connection* (**e**) $\sqsubset$ *Material connection* (**m**); *Communication by data* (**d**) $\sqsubset$ *Information connection* (**i**); *Control link* (**c**) $\sqsubset$ *Information connection* (**i**). From this hierarchy, rules are extracted to construct the concepts of an alphabetic set of nodes.

This article describes new ways of using an FSNS for system decomposition (building a partitive classification), taking into account the specific subject content of the area of analysis or design. The proposed approach simplifies the procedure of the graphic-analytical modeling of a complex system, limiting that role of developers' creative capabilities, which generates a variety of representations of organizational models, and thereby causes the least harm to the organization [6].

PECULIARITIES OF CONSTRUCTING A PARTITIVE CLASSIFICATION IN SODA

The main goal of any system analysis and the starting point of any system design is to determine the system-forming factor, i.e., why and for what the system exists or why and for what it was designed. G.P. Melnikov, the founder of determinant analysis emphasized that if, when studying a system, it was possible to first establish its external determinant (a functional request of a supersystem or a required functional state), then its current internal determinant (its actually existing functional state) and is derived from its external determinant by meaningful reasoning on the stages of system formation [12].

In SODA, the main goal is to establish a supersystem of an analyzed or designed system and a functional request for it, that is, to establish the cause of the system under consideration or its external determinant. This is achieved through the construction of a generic classification, i.e., hierarchies of classes (system-classes and conceptual systems), including the considered system-phenomenon (material system). Then, a stage (genetic) classification is constructed that describes the transition of a specific class (system-class), an instance of which is the considered system-phenomenon, into this instance in particular, that is, into this system-phenomenon [13, 14].

The result of the stage (genetic) classification of the system can be represented as a three-element construction of the system-object approach "Node–Function–Object," described in the language of descriptive logic [14]:

$$\begin{gathered}\mathbf{s}^i = [(\{LS?^i\} \sqcup \{LS\tau^i\} \sqcup \{LS!^i\}); \\ (\{LS!^i\} \sqcap \exists f.\{LS\tau^i\} \sqcap \exists f.\{LS?^i\}); \\ (\exists hasOS?^i = \mathbf{p1} \sqcup \exists hasOS\tau^i \\ = \mathbf{p2} \sqcup \exists hasOS!^i = \mathbf{p3})],\end{gathered}$$

where $\mathbf{s}^i$ is the system-phenomenon that is to be analyzed or designed; $(\{LS?^i\} \sqcup \{LS\tau^i\} \sqcup \{LS!^i\})$ is a concept for describing a system node $\mathbf{s}^i$ as an intersection of finite sets of input links $\{LS?^i\}$, internal links $\{LS\tau^i\}$ and output links $\{LS!^i\}$ in the supersystem structure; $(\{LS!^i\} \sqcap \exists f.\{LS\tau^i\} \sqcap \exists f.\{LS?^i\})$ is a concept for describing a system function $\mathbf{s}^i$, defined by a supersystem or a method that provides a functional correspondence between output $\{LS!^i\}$ and input $\{LS?^i\}$ flows of this node, taking into account the intermediate role (function) $f\tau$ of transformation of internal flows $\{LS\tau^i\}$; $(\exists hasOS? = \mathbf{k1} \sqcup \exists hasOS\tau = \mathbf{k2} \sqcup \exists hasOS! = \mathbf{k3})$ is a concept for describing the substantial (objective) characteristics of the system $\mathbf{s}^i$ (input, internal, and output), where **k1**, **k2**, and **k3** are attributes with specific values. Descriptive logic can be used to describe the decomposition procedure (construction of a partitive classification) of the system $\mathbf{s}^i$ into subsystems, the functional properties of which are supporting for the system $\mathbf{s}^i$, in the following way:

$$\mathbf{s}^{i,m} \sqsubset \mathbf{s}^i \ (m = 1,\dots,N); \text{ or } \mathbf{s}^i \equiv \mathbf{s}^{i,1} \sqcup \dots \sqcup \mathbf{s}^{i,N},$$

such that:

$$\begin{gathered}(\{LS!^{i,m}\} \sqcap \exists f\tau.\{LS\tau^{i,m}\} \\ \sqcap \exists f.\{LS?^{i,m}\}) \sqsubset \exists f\tau.\{LS\tau^i\}.\end{gathered}$$

Each subsystem $\mathbf{s}^{i,m}$, which in turn can be decomposed, will have the following form:

$$\begin{gathered}\mathbf{s}^{i,m} = [(\{LS?^{i,m}\} \sqcup \{LS\tau^{i,m}\} \sqcup \{LS!^{i,m}\}); \\ (\{LS!^{i,m}\} \sqcap \exists f.\{LS\tau^{i,m}\} \sqcap \exists f.\{LS?^{i,m}\}); \\ (\exists hasOS?^{i,m} = \mathbf{q1} \sqcup \exists hasOS\tau^{i,m} \\ = \mathbf{q2} \sqcup \exists hasOS!^{i,m} = \mathbf{q3})].\end{gathered}$$

FORMAL-SEMANTIC NORMATIVE SYSTEM

Consider the possibilities of the use of the FSNS to perform a decomposition procedure (building a partitive classification) for the system.

The construction of the rules for an alphabetical set of nodes (concepts) based on the hierarchy of concepts of links/flows defined in [8, 14] is briefly presented in the table, where the sign "**?**" corresponds to input connections/flows, and the sign "**!**" represents output. The classification of links can be refined by introducing the types of links **v**, **e**, **d**, and **c** and thereby expanded the range of alphabetic nodes. The depth of classification of links and, hence, the number and specific type of alphabetic nodes, are determined by the problem being solved. The rules for manipulating the symbols of an alphabet formed in this way correspond to the previously proposed rules of system composition [8, 9] and can be formalized by means of the apparatus of function calculus introduced in [15].

The use of a classification scheme ensures that the alphabet of the normative system is specified, which has not only a completely abstract or purely mathematical semantics, but is also subject (problem)-oriented, which allows us to consider this alphabet as formal-semantic (and, accordingly, the normative system itself is considered formal-semantic).

Methodologically, the FSNS can be used for the algorithmization and computer support of the decomposition procedure in a graphic-analytical modeling of complex objects, in two ways: first, using the "interface decomposition" proposed in [16]; second, the approach to automating the construction of system-object models in terms of "Node–Function–Object," proposed in [17]. We consider these methods on the example of modeling the production process, refining and supplementing the results obtained in [8, 9, 14].

Table 1. Rules for constructing the alphabet of FSNS nodes

No.	Sign	Formal description of the node	Subject content of the functional node as an element of transformation
1	**V**	$\mathbf{v}! \sqcap \exists$has a match. $(\mathbf{v}?)$	Substance
2	**E**	$\mathbf{e}! \sqcap \exists$has a match. $(\mathbf{e}?)$	Energy
3	**D**	$\mathbf{d}! \sqcap \exists$has a match. $(\mathbf{d}?)$	Data
4	**C**	$\mathbf{c}! \sqcap \exists$has a match. $(\mathbf{c}?)$	Management (control information)
5	**VE**	$(\mathbf{v}! \sqcup \mathbf{e}!) \sqcap \exists$has a match. $(\mathbf{v}? \sqcup \mathbf{e}?)$	Substances and energies
6	**VD**	$(\mathbf{v}! \sqcup \mathbf{d}!) \sqcap \exists$has a match. $(\mathbf{v}? \sqcup \mathbf{d}?)$	Substances and data
7	**VC**	$(\mathbf{v}! \sqcup \mathbf{c}!) \sqcap \exists$has a match. $(\mathbf{v}? \sqcup \mathbf{c}?)$	Substances and controls
8	**ED**	$(\mathbf{e}! \sqcup \mathbf{d}!) \sqcap \exists$has a match. $(\mathbf{e}? \sqcup \mathbf{d}?)$	Energy and data
9	**EC**	$(\mathbf{e}! \sqcup \mathbf{c}!) \sqcap \exists$has a match. $(\mathbf{e}? \sqcup \mathbf{c}?)$	Energy and control
10	**DC**	$(\mathbf{d}! \sqcup \mathbf{c}!) \sqcap \exists$has a match. $(\mathbf{d}? \sqcup \mathbf{c}?)$	Data and control

We represent the context diagram of the production process in the form of a node of the type **VD** (Fig. 1), which at the entrance are raw material ($\mathbf{v}?_{\text{raw material}}$) and source documents ($\mathbf{d}?_{\text{source\_doc}}$), and at the output are proven components ($\mathbf{v}!_{\text{pr\_comp}}$) and final documentation ($\mathbf{d}!_{\text{doc\_it}}$). The classification of links includes two types of real links and two types of data links: $\mathbf{v}_{\text{raw material}} \sqsubset \mathbf{v}$, $\mathbf{v}_{\text{pr\_comp}} \sqsubset \mathbf{v}$; $\mathbf{d}_{\text{source\_doc}} \sqsubset \mathbf{d}$, $\mathbf{d}_{\text{doc\_it}} \sqsubset \mathbf{d}$.

To decompose this process, it is necessary to determine the alphabetic elements that its internal structure consists of. If we restrict ourselves to the alphabet proposed above, we can assume three options for the internal structure:

(a) elements of type at the entrance and exit;

(b) at the input, there are two elements of type **V** and **D** at the output element of type;

(c) at the input element of type , at the output two elements of type **V** and **D**.

We consider option a) and show that further we can proceed in two ways.

(1) In the first case, using the interface decomposition [16], we add to the classification of links such links of the type **v** and **d**, which, on the one hand, are the output for the connections $\mathbf{v}?_{\text{raw material}}$ and $\mathbf{d}?_{\text{source\_doc}}$, and, on the other hand, as input for connections $\mathbf{v}!_{\text{pr\_comp}}$ and $\mathbf{d}!_{\text{doc\_it}}$. In this case, these may be details ($\mathbf{v}_{\text{details}} \sqsubset \mathbf{v}$) that are made from raw materials and from which proven components and intermediate documents can be produced ($\mathbf{d}_{\text{oc\_pr}} \sqsubset \mathbf{d}$) that can be obtained from source documents and final documents can be formed from them. Thus the node $\mathbf{VD_0}$ is divided into two functional (Fig. 2): $\mathbf{VD_1}$ (crossroads of connections $\mathbf{v}?_{\text{raw material}}$, $\mathbf{d}?_{\text{source\_doc}}$ and $\mathbf{v}!_{\text{details}}$, $\mathbf{d}!_{\text{doc\_pr}}$) and $\mathbf{VD_2}$ (crossroads of connections $\mathbf{v}?_{\text{details}}$, $\mathbf{d}?_{\text{doc\_pr}}$ and $\mathbf{v}!_{\text{pr\_comp}}$, $\mathbf{d}!_{\text{doc\_it}}$).

The received nodes $\mathbf{VD_1}$ and $\mathbf{VD_2}$ can be decomposed in a similar way. We add to the classification of links such links of the type **v** and **d**, which, on the one hand, are output for $\mathbf{v}?_{\text{raw material}}$ and $\mathbf{d}?_{\text{source\_doc}}$, and on the other hand, input for $\mathbf{v}!_{\text{details}}$, $\mathbf{d}!_{\text{doc\_pr}}$. In this case, these may be blanks ($\mathbf{v}_{\text{blank}} \sqsubset \mathbf{v}$), which are made from raw materials and parts can be produced from them and first internal documents ($\mathbf{d}_{\text{doc\_in1}} \sqsubset \mathbf{d}$) that can be obtained from source documents and intermediate documents can be formed from them. Thus the node $\mathbf{VD_1}$ is divided into two functional units: $\mathbf{VD_{11}}$ (crossroads of connections $\mathbf{??}_{\text{raw material}}$, $\mathbf{d}?_{\text{source\_doc}}$, $\mathbf{v}!_{\text{blank}}$, $\mathbf{d}!_{\text{doc\_in1}}$) and $\mathbf{VD_{12}}$ (crossroads of connections $\mathbf{v}?_{\text{blank}}$, $\mathbf{d}?_{\text{doc\_in1}}$ And $\mathbf{v}!_{\text{details}}$, $\mathbf{d}!_{\text{doc\_pr}}$).

We add to the classification of links such links of the type **v** and **d**, which, on the one hand, are the output for the connections $\mathbf{??}_{\text{details}}$, $\mathbf{d}?_{\text{doc\_pr}}$, and, on the other hand, are input for $\mathbf{v}!_{\text{pr\_comp}}$ and $\mathbf{d}!_{\text{doc\_it}}$. In this case, these may be accessories ($\mathbf{v}_{\text{comp}} \sqsubset \mathbf{v}$) that can be produced from parts and from which tested components can be produced, and second internal documents ($\mathbf{d}_{\text{doc\_in2}} \sqsubset \mathbf{d}$) that can be obtained from intermediate documents and final documents can be formed from them. Thus the node $\mathbf{VD_2}$ is divided into two functional units: $\mathbf{VD_{21}}$ (crossroads of connections $\mathbf{v}?_{\text{details}}$, $\mathbf{d}?_{\text{doc\_pr}}$ and $\mathbf{v}!_{\text{comp}}$, $\mathbf{d}!_{\text{doc\_in2}}$) and $\mathbf{VD_{22}}$ (cross-

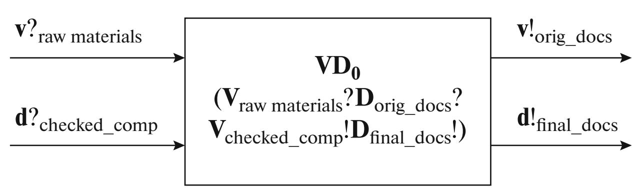

Fig. 1. Context diagram of the manufacturing process $\mathbf{VD_0}$.

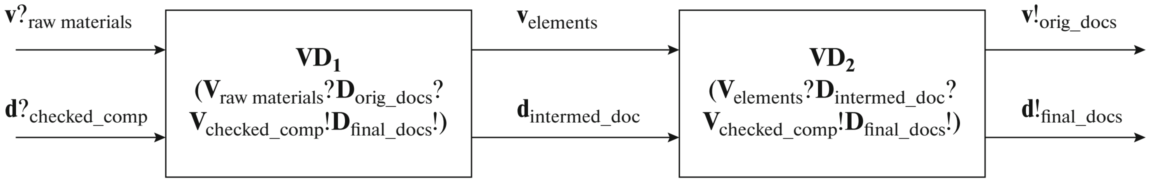

Fig. 2. Diagram 1 of the decomposition of the production process $\mathbf{VD_0}$.

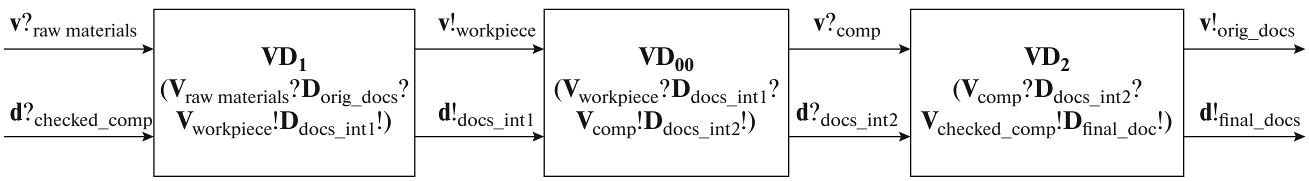

Fig. 3. Diagram 2 of the decomposition of the production process $\mathbf{VD_0}$.

roads of connections $\mathbf{v}?_{comp}$, $\mathbf{d}?_{doc\_in2}$ and $\mathbf{v}!_{pr\_comp}$, $\mathbf{d}!_{doc\_it}$).

The final decomposition (partitive classification) contains four sequentially connected nodes of the type **VD**. In terms of the function calculus proposed in [15], the decomposition results can be written as the result of the "prefix action" operation:

$$\mathbf{VD_0} = \mathbf{VD}?_1 \,.\, \mathbf{VD_2} = \mathbf{VD}?_{11} \,.\, \mathbf{VD}?_{12} \,.\, \mathbf{VD}?_{21} \,.\, \mathbf{VD_{22}};$$

or as a result of the "postfix action" operation:

$$\mathbf{VD_0} = \mathbf{VD}!_2 \,.\, \mathbf{VD_1} = \mathbf{VD}!_{22} \,.\, \mathbf{VD}!_{21} \,.\, \mathbf{VD}!_{12} \,.\, \mathbf{VD_{11}}.$$

(2) In the second case, using the approach proposed in [17], we add to the classification of links of the type **v** and **d**, which, on the one hand, are direct output streams for links $\mathbf{v}?_{raw\ material}$ and $\mathbf{d}?_{source\_doc}$, and, on the other hand, direct input flows for connections $\mathbf{v}!_{pr\_comp}$ and $\mathbf{d}!_{doc\_it}$. In this case, on the one hand, it will be blanks ($\mathbf{v}_{blank} \sqsubset \mathbf{v}$), that can be made directly from raw materials and first internal documents ($\mathbf{d}_{doc\_in1} \sqsubset \mathbf{d}$) directly obtained from source documents. On the other hand, it will be accessories ($\mathbf{v}_{comp} \sqsubset \mathbf{v}$) from which proven components can be produced, and second internal documents ($\mathbf{d}_{doc\_in2} \sqsubset \mathbf{d}$), from which final documents can be formed. Thus the node $\mathbf{VD_0}$ is divided into three functional units (Fig. 3): $\mathbf{VD_1}$ (crossroads of connections $\mathbf{v}?_{raw\ material}$, $\mathbf{d}?_{source\_doc}$ and $?!_{blank}$, $\mathbf{d}!_{doc\_in1}$), $\mathbf{VD_{00}}$ (crossroads of connections $\mathbf{v}?_{blank}$, $\mathbf{d}?_{doc\_in1}$ and $\mathbf{v}!_{comp}$, $\mathbf{d}!_{doc\_in2}$) and $\mathbf{VD_2}$ (crossroads of connections $\mathbf{v}?_{comp}$, $\mathbf{d}?_{doc\_in2}$ and $\mathbf{v}!_{pr\_comp}$, $\mathbf{d}!_{doc\_it}$).

As a result, the problem of the decomposition of the production process is reduced to the previous one, but with other input and output connections that can be solved both in the first and in the second way. Taking into consideration the previous classification of links, the node $\mathbf{VD_{00}}$ easily split into two functional units $\mathbf{VD_{01}}$ (crossroads of connections $\mathbf{v}?_{blank}$, $\mathbf{d}?_{doc\_in1}$ and $\mathbf{v}!_{details}$, $\mathbf{d}!_{doc\_pr}$) and $\mathbf{VD_{02}}$ (crossroads of connections $\mathbf{v}?_{details}$, $\mathbf{d}?_{doc\_pr}$ and $\mathbf{v}!_{comp}$, $\mathbf{d}!_{doc\_in2}$).

Both methods give the same final decomposition in the form of a chain of the same link intersections (nodes).

ALGORITHM FOR USING THE FSNS

The algorithm for the use of a formal-semantic normative system in constructing a partitive classification of the system includes the following steps in its first version.

Step 1. Let $\mathbf{s}^i = [(\{LS?^i\} \sqcup \{LS\tau^i\} \sqcup \{LS!^i\}); (\{LS!^i\} \sqcap \exists f.\{LS\tau^i\} \sqcap \exists f.\{LS?^i\}); (\exists hasOS?^i_{=}\ \mathbf{p1} \sqcup \exists hasOS\tau^i = \mathbf{p2} \sqcup \exists hasOS!^i = \mathbf{p3})]$ be the system-phenomenon to be analyzed or designed. Let NS be a formal-semantic normative system (FSNS), consisting of the basic classification of links and nodes mentioned earlier. We perform the initial classification of the links of the original system-phenomenon $\mathbf{s}^i$, using the FSNS for each input and output, where $\{LS?^i\} \equiv L?^i_1 \sqcup L?^i_{2\ldots} \sqcup L?^i_N$ and $\{LS!^i\} \equiv L!^i_1 \sqcup L!^i_{2\ldots} \sqcup L!^i_N$ that $\exists$LinkType.$\{L?^i_j\} \sqsubset NS$, (j = 1..N) and $\exists$LinkType.$\{L!_j\} \sqsubset NS$, (j = 1..N), depending on the chosen type of connec-

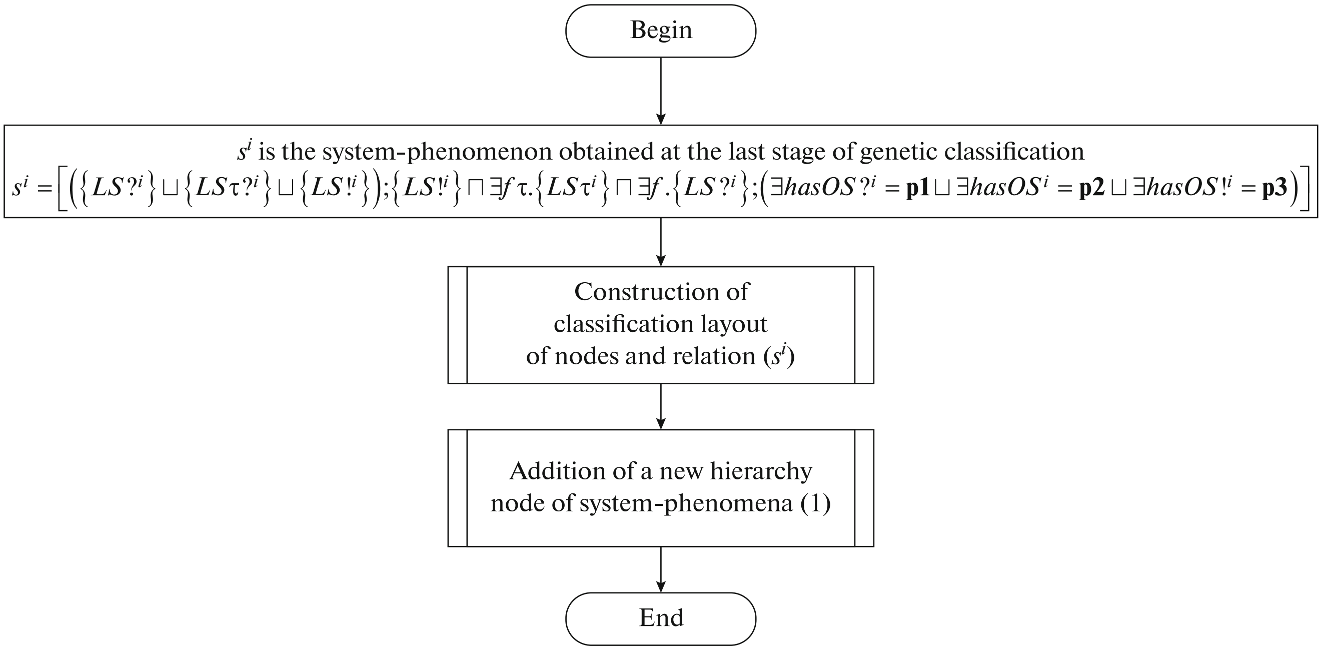

Fig 4. Flowchart of partitive classification using FSNS.

tions, we define the system $\mathbf{s}^i$, as one of the alphabetic nodes of the normative system: AlphabeticalKnot^i ≡ (∃LinkType.$\{L!^i_j\}$ ⊔ ... ⊔ ∃LinkType.$\{L!^i_N\}$) ⊓ ∃has a match. (∃LinkType.$\{L?^i_j\}$⊔ ... ⊔ ∃LinkType.$\{L?^i_N\}$); AlphabeticalKnot$^i \sqsubset NS$.

Step 2. We add to the classification of links such links L^{i+1}_j related to the type of connections of the supersystem, which ∃LinkType.$\{L^{i+1}_j\}$ = ∃LinkType.$\{L?^i_j\}$ = ∃LinkType.$\{L!^i_j\}$, which are output for ties $L?^i_j$ and input for connections $L!^i_j$.

Step 3. We perform decomposition AlphabeticalKnot^i on two functional units AlphabeticalKnot$^{i+1,1}$ and AlphabeticalKnot$^{i+1,2}$, wherein AlphabeticalKnot$^{i+1,1}$ ≡ (∃LinkType.$\{L^i_j\}$ ⊔ ... ⊔ ∃LinkType.$\{L^i_N\}$) ⊓ ∃has a match. (∃LinkType.$\{L?^i_j\}$⊔ ... ⊔ ∃LinkType.$\{L?^i_N\}$); and AlphabeticalKnot$^{i+1,2}$ ≡ (∃LinkType.$\{L!^i_j\}$ ⊔ ... ⊔ ∃LinkType.$\{L!^i_N\}$) ⊓ ∃has a match. (∃LinkType.$\{L^i_j\}$⊔ ... ⊔ ∃LinkType.$\{L^i_N\}$), ($j = 1..N$).

Step 4. Repeat Step 2 until the alphabetic nodes corresponding to all input and output links of the supersystem are entered $\mathbf{s}^i$.

Step 5. We perform system decomposition for $\mathbf{s}^i$ into subsystems, the functional properties of which are supporting for the system, as follows: $\mathbf{s}^{i,m} \sqsubset \mathbf{s}^i$ ($m = 1, N$); or $\mathbf{s}^i \equiv \mathbf{s}^{i,1} \sqcup ... \sqcup \mathbf{s}^{i,N}$, such that: $(\{LS!^{i,m}\} \sqcap \exists f\tau.\{LS\tau^{i,m}\} \sqcap \exists f.\{LS?^{i,m}\}) \sqsubset \exists f\tau.\{LS\tau^i\}$. In other words, for each subsystem $s^{i,m}$ we reveal its structure based on the nodes and links introduced in Steps 2–4 in NS: $\mathbf{s}^{i,m} = [(\{LS?^{i,m}\} \sqcup \{LS\tau^{i,m}\} \sqcup \{LS!^{i,m}\}); (\{LS!^{i,m}\} \sqcap \exists f.\{LS\tau^{i,m}\} \sqcap \exists f.\{LS?^{i,m}\}); (\exists hasOS?^{i,m} = \mathbf{q1} \sqcup \exists hasOS\tau^{i,m} = \mathbf{q2} \sqcup \exists hasOS!^{i,m} = \mathbf{q3})]$.

Step 6. Repeat Step 5 for all subsystems of systems-phenomena $\mathbf{s}^{i,m}$.

The second option for using the FSNS differs in Steps 2 and 3, which are given below:

• We add such links to the classification of links $L?^{i+1}_j$ and $L!^{i+1}_j$ related to the type of connections of the supersystem, which ∃LinkType.$\{L?^{i+1}_j\}$ = ∃LinkType.$\{L?^i_j\}$ and ∃LinkType.$\{L!^{i+1}_j\}$ = ∃LinkType.$\{L!^i_j\}$, So $L?^{i+1}_j$ are the output for connections $L?^i_j$, $L!^{i+1}_j$ and input for $L!^i_j$;

• AlphabeticalKnot^i is decomposed on three functional nodes AlphabeticalKnot$^{i+1,1}$, AlphabeticalKnot$^{i+1,2}$, and AlphabeticalKnot$^{i+1,3}$, wherein AlphabeticalKnot$^{i+1,1}$ ≡ (∃LinkType.$\{L?^{i+1}_j\}$ ⊔ ... ⊔ ∃LinkType.$\{L?^{i+1}_N\}$) ⊓ ∃has a match. (∃LinkType.$\{L?^i_j\}$⊔ ... ⊔ ∃LinkType.$\{L?^i_N\}$); AlphabeticalKnot$^{i+1,2}$ ≡ (∃LinkType.$\{L!^{i+1}_j\}$ ⊔ ... ⊔ ∃LinkType.$\{L!^{i+1}_N\}$) ⊓ ∃has a match. (∃LinkType.$\{L?^{i+1}_j\}$⊔ ... ⊔ ∃LinkType.$\{L?^{i+1}_N\}$); AlphabeticalKnot$^{i+1,3}$ ≡ (∃Link-

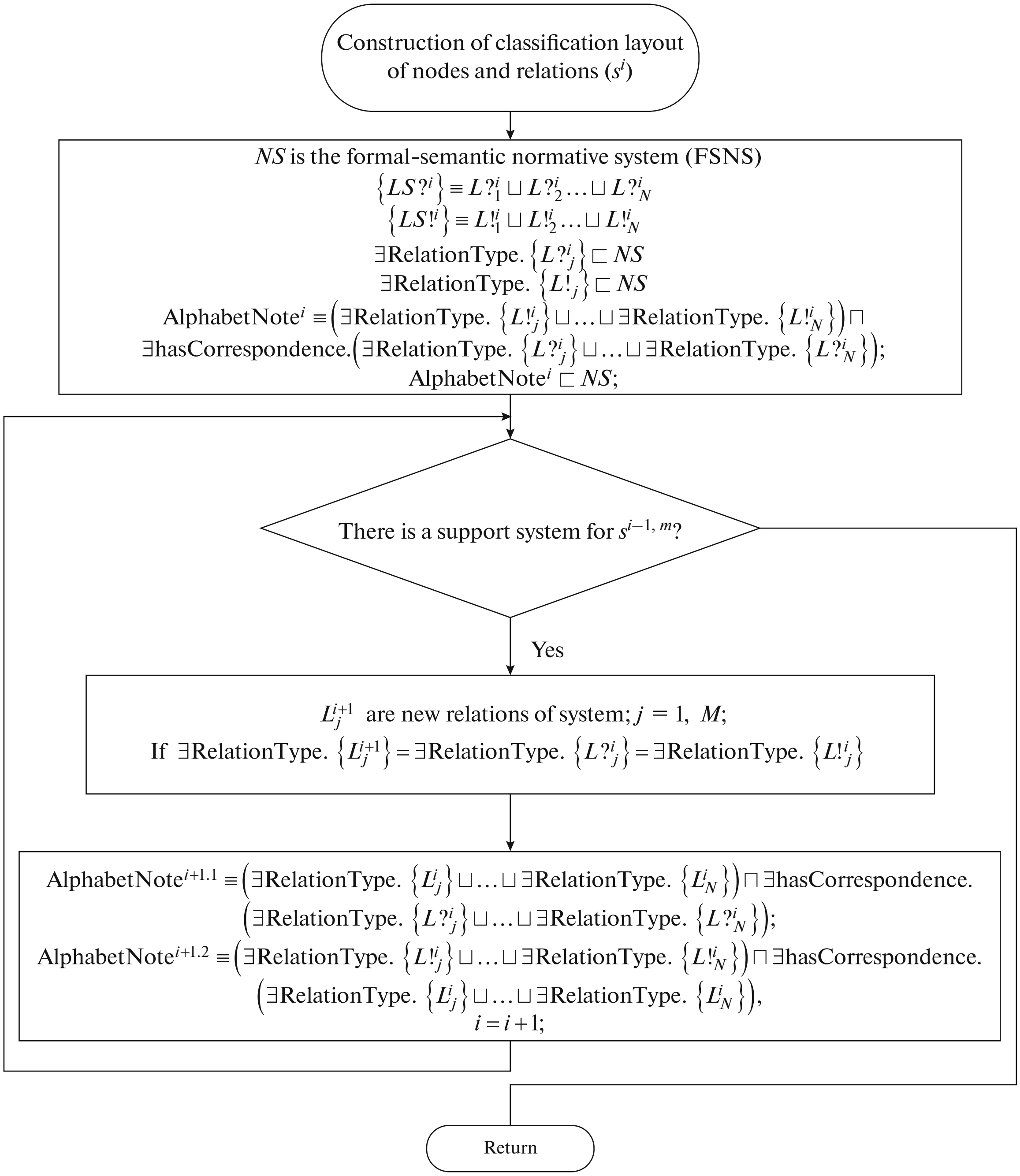

Fig. 5. Block diagram of the partitive classification according to the first method of using the FSNS.

Type.$\{L!_j^i\} \sqcup \ldots \sqcup \exists$LinkType.$\{L!_N^i\}) \sqcap \exists$has a match. ($\exists$LinkType.$\{L?_j^{i+1}\} \sqcup \ldots \sqcup \exists$LinkType.$\{L?_N^{i+1}\}$); ($j$ = 1..N).

An extended block diagram of the algorithm for constructing a partitive classification [14] is shown in Fig. 4, and flowcharts of the algorithms of the first and second methods of using the FSNS are shown in Figs. 5 and 6.

CONCLUSIONS

This work completes a series of articles presenting the results of the development of a system-object determinant analysis that is designed to provide computer support for the analysis of complex applied research objects. SODA is distinguished by the algorithmic support for all stages and processes, as well as the ability to take into account known system-wide patterns in the analysis.

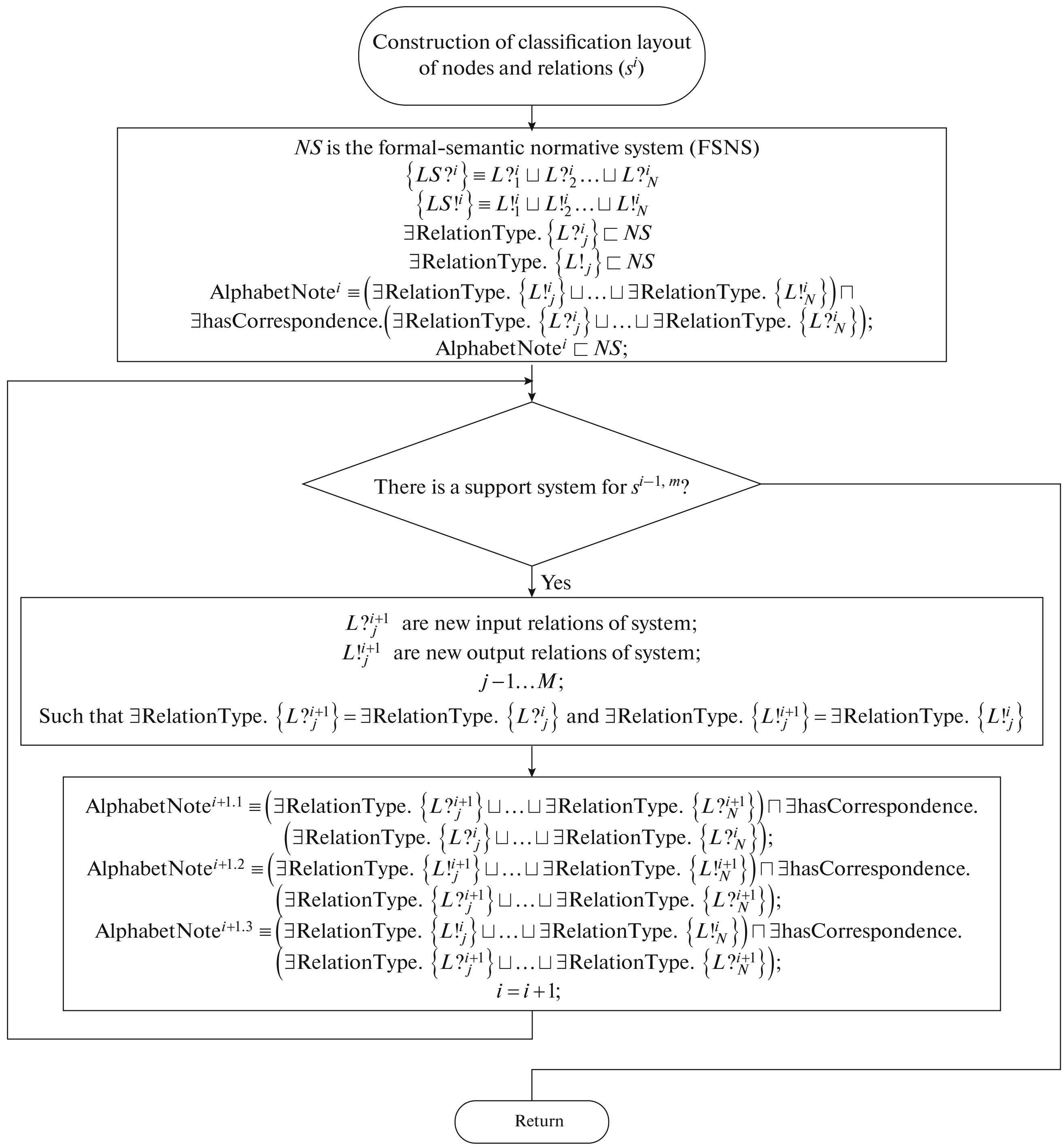

Fig. 6. Block diagram of partitive classification according to the second method of using the FSNS.

The use of a formal-semantic normative system for the construction construct of a partitive classification (decomposition) of a system at the final stage of SODA simplifies the procedures for the graphic-analytical modeling of a complex system. In addition, the use of FSNS makes it possible to take into account the substantive aspects of the area of analysis without losing the rigor and accuracy of the description of the object under study.

FUNDING

The work is supported by the Russian Foundation for Basic Research, project no. 19-29-01047mk.

CONFLICT OF INTEREST

The authors of this work declare that they have no conflicts of interest.

REFERENCES

1. Dubeikovskii, V.I., *Effektivnoe modelirovanie s AllFusion Process Modeler i AllFusion PM* (Effective Modeling in AllFusion Process Modeler and AllFusion PM), Moscow: Dialog-MIFI, 2007.
2. Vendrov, A.M., *Proektirovanie programmnogo obespecheniya ekonomicheskikh informatsionnykh sistem* (Design of Software of Economic Information Systems), Moscow: Finansy i Statistika, 2000.
3. Maklakov, S.V., *Modelirovanie biznes-protsessov s AllFusion Process Modeler* (Modeling of Business Processes with AllFusion Process Modeler), Moscow: Dialog-MIFI, 2004.
4. Elizarova, N.N. and Arkhangel'skaya, E.L., Application of graphoanalytic method for analyzing subject area in designing information systems, *Vestn. Ivanovsk. Gos. Energ. Univ.*, 2010, no. 4, pp. 106–110.
5. Kaziev, V.M., *Vvedenie v analiz, sintez i modelirovanie sistem* (Introduction to Analysis, Synthesis, and Modeling of Systems), Moscow: Prosveshchenie/Binom, 2014.
6. Rubtsov, S., With CASE tool make the least harm to an enterprise?, *Direktor IS*, 2002, no. 1. http://www.osp.ru/cio/2002/01/008.htm.
7. Matorin, S.I. and Mikhelev, V.V., A system–object approach to determinant analysis of complex systems, *Sci. Tech. Inf. Process.*, 2021, vol. 48, no. 5, pp. 327–332. https://doi.org/10.3103/S014768822105004X
8. Matorin, S.I., Mikhelev, V.V., and Zhikharev, A.G., Normative system of system-object analysis and modeling, *Ekon. Inf.*, 2020, no. 3, pp. 623–637.
9. Matorin, S.I. and Mikhelev, V.V., Formal-semantic normative system for graphic-analytical modelling, *J. Phys.: Conf. Ser.*, 2021, vol. 2060, no. 1, p. 012020. https://doi.org/10.1088/1742-6596/2060/1/012020
10. Baader, F., Calvanese, D., McGuinness, L., Nardi, D., and Patel-Schneider, P., *The Description Logic Handbook: Theory, Implementation, and Application*, Cambridge Univ. Press, 2003. https://doi.org/10.1017/cbo9780511711787
11. Baader, F. and Sattler, U., Expressive number restrictions in description logics, *J. Logic Comput.*, 1999, vol. 9, no. 3, pp. 319–350. https://doi.org/10.1093/logcom/9.3.319
12. Mel'nikov, G.P., *Sistemologiya i yazykovye aspekty kibernetiki* (Systemoloby and Language Aspects of Cybernetics), Moscow: Sov. Radio, 1978.
13. Matorin, S.I. and Mikhelev, V.V., System–object determinant analysis: Constructing a domain taxonomy, *Sci. Tech. Inf. Process.*, 2022, vol. 49, no. 5, pp. 325–332. https://doi.org/10.3103/S0147688222050069
14. Matorin, S.I. and Mikhelev, V.V., System–object determinant analysis: Construction of genetic and partitive classification of the subject area, *Sci. Tech. Inf. Process.*, 2023, vol. 50, no. 6.
15. Zimovets, O.A. and Matorin, S.I., Integration of formalization tools for graphical-analytical "unit–function–object" models, *Sci. Tech. Inf. Process.*, 2013, vol. 40, no. 6, pp. 396–402. https://doi.org/10.3103/S0147688213060099
16. Matorin, S.I. and Zimovets, O.A., Formalized visual modeling of administrative procedures, *Prikl. Inf.*, 2012, no. 2, pp. 100–110.
17. Matorin, S.I., Zimovets, O.A., and Zhikharev, A.G., On the technology development of graphic-analytical business modeling with the use of the system approach of "knot–function–object," *Autom. Doc. Math. Linguist.*, 2007, vol. 41, no. 6, pp. 242–250. https://doi.org/10.3103/S0005105507060027

Publisher's Note. Allerton Press remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.