

# The graph template for designing the conceptual and logical model of business processes and technical systems

Valerii P. Khranilov, Pavel V. Misevich, Pavel S. Kulyasov, and Elena N. Pankratova

**Abstract**— Currently the task of reducing the cost of software and information support for processes in various subject areas is actual. The method of developing a conceptual model in the form of a graph-template is described in the article. The proposed method allows IT designers to create conceptual models with minimal cost. The use of the template graph allows one to create a graph conceptual model that is mapped into a multidimensional logical data schema. The proposed method is oriented at developing a multidimensional database schema. The method is based on a sequence of steps: the research of the subject area; the creation of process models; the design of a conceptual model; the design of a multidimensional database model. The multidimensional model supports the use of business intelligence methods and allows IT staff to use OLAP analysis. The technology to adapt the graph-template for describing a subject area is used for developing the conceptual model. Then the conceptual graph model transforms to a multidimensional data model for supporting OLAP analysis. Thus, the use of a graph template reduces the cost of designing information support using DB. This method has been tested in the design of information support of business processes and monitoring systems.

**Keywords**— *business process model, EPC, frame, technical systems.*

## I. INTRODUCTION

Currently the task of reducing the cost of software and information support for processes in various subject areas is actual.

The relevance of the task is confirmed by the development of various new IT technologies. They permanently improve the software and information support of business and technical processes, reduce overall costs of IT projects, and improve the design process.

We highlight the use of templates for building software [1] and information support [2] of processes as a promising area of research. The paper [2] describes a method for building information support for business processes in various subject areas. The method uses templates of the conceptual model. The model is based on the abstract graph

template.

The method is based on adapting the elements of an abstract graph template to a specific subject area. The adaptation involves setting associative links between the business process model and template elements.

They are the following:

- sets of edges;
- vertices;
- typical classes of objects that are "attached" to graph elements, if it is necessary.

Thus, the graph template is transformed into a graph conceptual description of the subject area. The conceptual graph model is mapped into a multidimensional data model or implemented using a snowflake structure in a relational database.

## II. TECHNOLOGY FOR DEVELOPING INFORMATION SUPPORT

Let us define the categories that describe the technology for designing a multidimensional model.

**Definition 1.** A conceptual model template is a data structure that is "customized" to a specific domain and transformed into a conceptual data model.

**Definition 2.** A graph template of a conceptual model is an abstract graph that keeps a set of objects. They are the following:

- vertexes;
- edge of the graph;
- theoretical primitives of various types that are "attached" to graph elements to detail the description of the subject area.

**Definition 3.** The adaptation of the graph template to the subject area consists of a sequence of steps. They are the following:

- defining a set of specific vertices in a conceptual model (the total number of vertices which can be divided into subsets);
- setting associative links between vertices and elements of the subject area;
- defining the set of edges (the total number of edges, dividing them into subsets, if it is necessary);
- formalizing the general structure of the graph-conceptual model (for example, using the incidence matrix);
- detailing the conceptual model to assign the elements of the theoretical primitives (definition 2) to the graph – conceptual model. The associative links are used for assignments.

Manuscript received March 10, 2026.

Valerii P. Khranilov, Nizhny Novgorod State Technical University (e-mail: khranilov@nntu.ru).

Pavel V. Misevich, Nizhny Novgorod State Technical University, Nizhny Novgorod (e-mail: p\_misevich@mail.ru).

Pavel S. Kulyasov, Nizhny Novgorod State Technical University, Nizhny Novgorod (e-mail: p.kulyasov@gmail.com).

Elena N. Pankratova, Nizhny Novgorod State Technical University, (e-mail: keibusan@gmail.com).

Definition 4. The template graph is mapped into a multidimensional data model using two steps. Step 1. Determine the number of dimensions (axes) and assign names to them. Step 2. Distribute the set of facts in the cells of the multidimensional model.

The method of building information support is based on the three stages. They are the following.

1. The study of the subject area and the development of the model (of a process or a system) using notations that are natural for specialists, users and experts in the field. The result of this stage is the formalization of the model: the definition of a system of objects and relationships between them.
2. The graph template is superimposed on the structure of the model (step 1). Thus, the graph template adapts to the subject area and transforms into the graph conceptual model.
3. The conceptual model is mapped onto the multidimensional data model (definition 4).

Let us describe the composition of the template graph. The template graph consists of a set of vertices  $V = \{v1, v2, \dots, vk\}$  and a set of edges  $E = \{e1, e2, \dots, el\}$ .

Objects of various types can be attached to the set of edges and vertices (theoretical primitives – definition 2). The objects can be the following:

- numerical values (measures);
- descriptors to various types of computational procedures (arithmetic, logical and others);
- frames i.e. tools used for describing situations;
- comments;
- sensors;
- other elements are primitives from the subject area of the study (for example, agents, etc.).

The conceptual model supports attaching various types of procedures to vertices and edges directly (without frames or intermediate objects).

It is also possible to attach comments to the vertices or edges of the graph. We do not consider the commentary as a frame, because it does not have the tools to adapt to the subject area.

Let us consider the frame as the main element, as theoretical primitive, which is used to modify a mathematical graph. The frame structure can be described by the Backus-Naur form.

```
frame ::= <classic frame by M. Minsky>
| <multimedia frame> | <frame with fuzzy logic
  procedures>
```

The frame consists of a name (ID), an unlimited number of slot - terminal pairs, and procedures i.e. demons [3].

Practice has shown that it is useful to attach multimedia objects [4] or procedures with fuzzy logic [5] to frames.

The multimedia frames [6] are an extension of the concept of a frame, which was introduced by Marvin M. Minsky. The multimedia frame may include multimedia objects.

```
Multimedia object ::= <electronic map>
| <presentation> | <diagram> | <video clip>
| <animation> | <audio track>
```

The analysis of technical tasks in the subject areas of monitoring and remote control systems [7] has shown the

expediency of using multimedia frames in the practice of an IT designer.

### III. EXAMPLE

The process [8] is described in the notation of the EPC diagram. It describes the life cycle of the request (Fig. 1). The description of the EPC diagram blocks is presented in Table 1. Let us develop a conceptual model (CM) for collecting the data about the passage of the requests in the company. This data is used for statistical analysis of customer service quality. The conceptual model is developed in the form of a graph (Fig. 2). In the graph the vertices correspond to events, functions, and logical operators (Fig. 1). Frames are attached to the vertices of the graph to collect statistics. The list of frames is shown in Table 1.

All vertices of the conceptual model are associated in the subject area with the events in the EPC diagram.

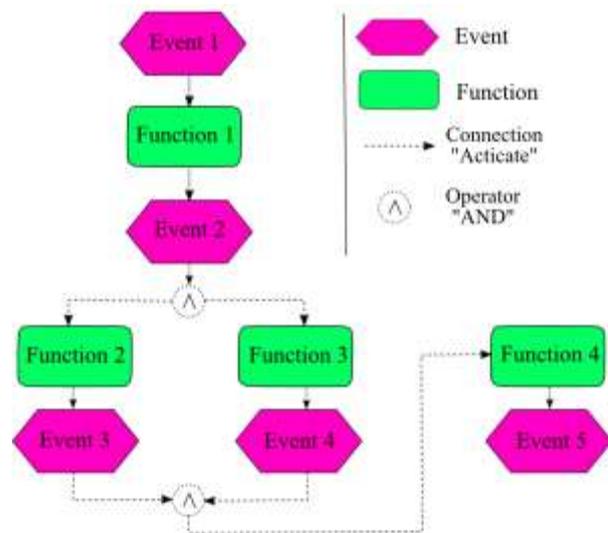


Fig. 1. Example of a business process model in EPC diagram notation.

TABLE I. THE DESCRIPTION OF THE EPC DIAGRAM BLOCKS

Block	Description
Event 1	A request for connection of the service has been received
Function 1	Register the request
Event 2	The request has been registered
Function 2	Issue a contract
Function 3	Organize the connection
Event 3	The contract has been signed
Event 4	The connection is organized
Function 4	Start providing the service
Event 5	The client's request has been processed

The article [9] describes the process of mapping a graph – conceptual model into a logical database schema in the form of a "snowflake" (into a multidimensional model). This is a typical transformation of a conceptual graph model into a logical schema of a multidimensional model. The logical schema is implemented in the MS SQL Server database

management system. Here is a logical diagram of the database which is based on the conceptual graph model (Fig. 2).

TABLE II. LIST OF FRAMES FOR COLLECTING STATISTICS ON REQUEST AND ON THE STAGES OF THEIR LIFE CYCLE

ID of the graph vertex OR frame	Slot name	Terminal
V1	Slot 1 - request cycle ID Slot 2 - Date and time of receipt Slot 3 - The type of request is the customer's status ID	Terminal 1 keeps an integer identifier Terminal 2 – date-time Terminal 3 is an integer identifier of the customer's status.
V2	Slot 1 - Date time:	Function 1 Terminal 1 – Date and time of receipt
V3	Slot 1 - Date time:	Terminal 1 – Date and time of receipt
V4	Slot 1 - Date time:	Terminal 1 – Date and time of receipt
V5	Slot 1 - Date time:	Terminal 1 – Date and time of receipt
V6	Slot 1 - Date time:	Terminal 1 – Date and time of receipt
V7	Slot 1 - Date time:	Terminal 1 – Date and time of receipt
V8	Slot 1 - Date time:	Terminal 1 – Date and time of receipt
V9	Slot 1 - Date time:	Terminal 1 – Date and time of receipt
V10	Slot 1 - Date time:	Terminal 1 – Date and time of receipt
V11	Slot 1 - Date time:	Terminal 1 – Date and time of receipt

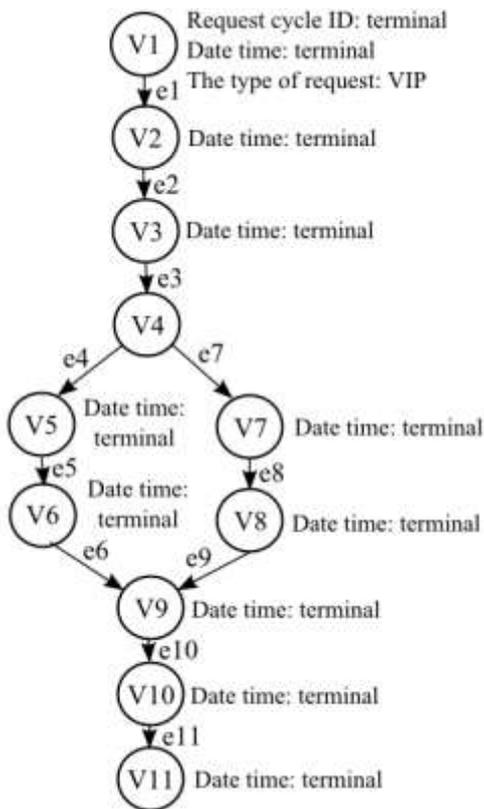


Fig. 2. Model in the form of the graph.

Let us describe the tables. The CD table stores the incidence matrix (Fig. 3). The tables named V and E are the catalogs of vertices and edges. The list of vertex classes is

stored in the table named Type\_V. The catalog of frame identifier names is stored in the table named F. For each frame, the set of pairs (slot ID) is stored in the S-T table.

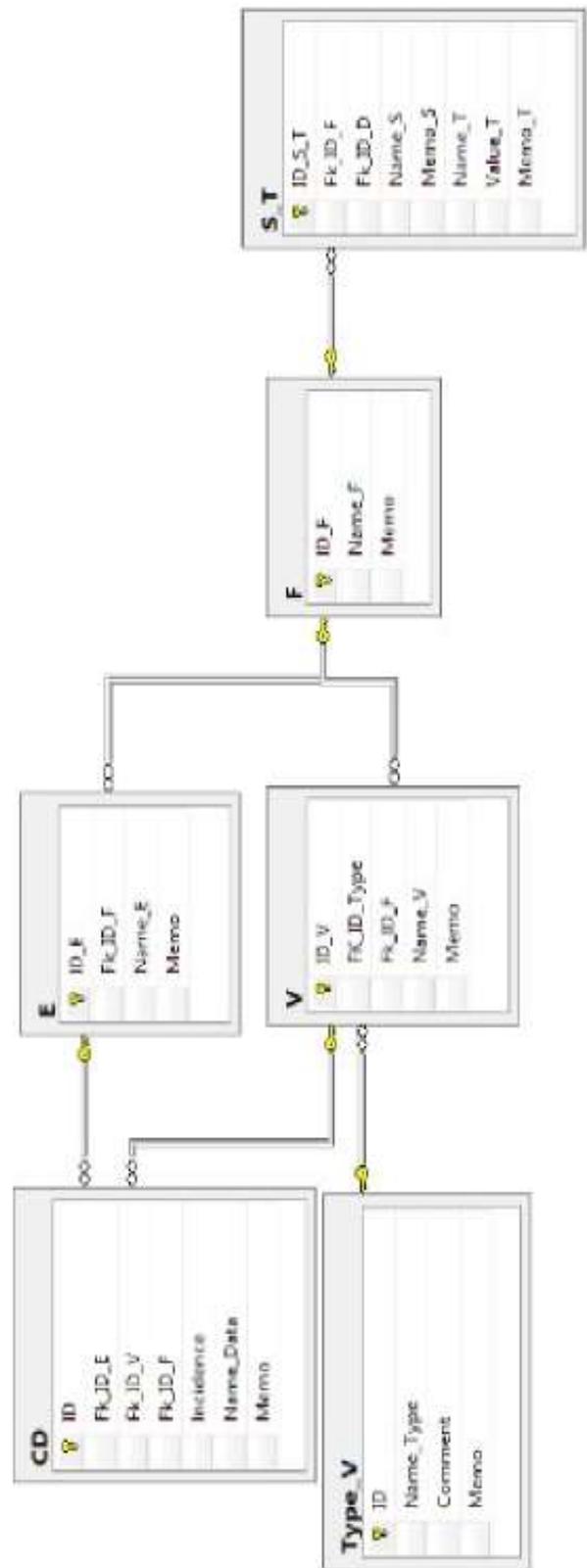


Fig. 3. Implementation in the logical scheme of the conceptual model.

Let us focus on the logical data model (Fig. 3). The tables named V and E are the catalogs of vertices and edges. Let us consider the types of fields in Table V. The fields are described in Table 3. The table stores the description of 11 vertices in the graph (Fig. 2) in our example.

TABLE III. THE DESCRIPTION OF THE FIELDS IN TABLE V

№	Column name	Data type	Note
1	ID_V	int	Key field
2	FK_ID_Type	int	Foreign key. The key is used to identify the vertex class
3	Fk_ID_F	int	Foreign key. The key is used to identify the ID frame
4	Name	nchar(50)	The field stores the vertex name
5	Memo	nvarchar (MAX)	The field keeps a comment

Table 4 shows the rows of the table V. The columns describe the vertices of the example graph.

TABLE IV. THE ROWS OF THE TABLE V

ID_V	FK_ID_Type	Fk_ID_F	Name_V
1	1	1	V1
2	2	2	V2
3	1	3	V3
4	3	4	V4
5	2	5	V5
6	1	6	V6
7	2	7	V7
8	1	8	V8
9	3	9	V9
10	2	10	V10
11	1	11	V11

The table *Type\_V* stores information about vertex classes in the conceptual model. Table 5 describes the names of its fields and their formats.

TABLE V. THE DESCRIPTION OF THE FIELDS IN TABLE TYPE\_V

№	Column name	Data type	Note
1	ID_V	int	Key field
2	Name_Type	nvarchar (50)	Foreign key. The key is used to identify the vertex class
3	Comment	nvarchar (MAX)	The field keeps a comment

Table 6 shows the fields in the table *Type\_V*.

TABLE VI. THE FIELDS IN TABLE TYPE\_V

ID	Name_Type	Comment
1	Event	NULL
2	Function	NULL
3	Logical operator	NULL

The fields of the table *E* are described in Table 7. The table stores the description of the graph edges (Fig. 2).

TABLE VII. THE DESCRIPTION OF THE FIELDS IN TABLE E

№	Column name	Data type	Note
1	ID_E	int	Key field
2	Fk_ID_F	int	Foreign key. The key is used to identify the ID frame
3	Name	nchar(50)	The field stores the edge name
4	Memo	nvarchar (MAX)	The field keeps a comment

Table 8 shows the columns in table *E*. The columns describe the edges of the graph (Fig. 2).

TABLE VIII. THE COLUMNS IN TABLE E

ID	Name_Type	Comment
1	E1	Between V1 and V2
2	E2	Between V2 and V3
3	E3	Between V3 and V4
4	E4	Between V4 and V5
5	E5	Between V5 and V6
6	E6	Between V6 and V9
7	E7	Between V4 and V7
8	E8	Between V7 and V8
9	E9	Between V8 and V9
10	E10	Between V9 and V10
11	E11	Between V10 and V11

Let's describe the fields in the table *CD* (table 9).

TABLE IX. THE DESCRIPTION OF THE FIELDS IN TABLE CD

№	Column name	Data type	Note
1	ID_V	int	Key field
2	Fk_ID_E	int	Foreign key. The key is used to identify the vertex class
3	Fk_ID_F	int	Foreign key. The key is used to identify the ID frame
4	Incidence	int	The field stores the value of +1 if the edge enters the vertex, and -1 if the edge exits the vertex
5	Name_Data	nvarchar (50)	The field stores the incident name
6	Memo	nvarchar (MAX)	The field keeps a comment

The fragment of the graph description (Fig. 2) in the *CD* table is presented in the table 10.

The table *F* stores information about the frame name. Table 11 stores a description of its fields.

Table 12 describes the rows of columns in table *F*.

The *S\_T* table stores information about the set of slot-terminal pairs. The number of the pairs is unlimited. The set of pairs defines a description of the situation in the subject

area. Table 13 describes the names and fields in the *S\_T* table.

TABLE X. THE FRAGMENT OF THE GRAPH DESCRIPTION IN THE *CD* TABLE

ID	Fk_ID_E	Fk_ID_V	Incidence
1	1	1	-1
2	1	2	1
3	2	2	-1
4	2	3	1
5	3	3	-1
6	3	4	1
7	4	4	-1
8	4	5	1
9	5	5	-1
10	5	6	1
11	6	6	-1
12	6	9	1
13	7	4	-1
14	7	7	1
15	8	7	-1
16	8	8	1
17	9	8	-1
18	9	9	1
19	10	9	-1
20	10	10	1
21	11	10	-1
22	11	11	1

TABLE XI. THE DESCRIPTION OF THE FIELDS IN TABLE *F*

№	Column name	Data type	Note
1	ID_F	int	Key field
2	Name_F	nvarchar (50)	Frame Name
3	Memo	nvarchar (MAX)	The field keeps a comment

TABLE XII. THE ROWS OF COLUMNS TABLE *F*

ID_F	Name_F
1	monitoring events of the vertex V1
2	monitoring events of the vertex V2
3	monitoring events of the vertex V3
4	monitoring events of the vertex V4
5	monitoring events of the vertex V5
6	monitoring events of the vertex V6
7	monitoring events of the vertex V7
8	monitoring events of the vertex V8
9	monitoring events of the vertex V9
10	monitoring events of the vertex V10

ID_F	Name_F
11	monitoring events of the vertex V11

TABLE XIII. THE DESCRIPTION OF THE FIELDS IN TABLE *S\_T*

№	Column name	Data type	Note
1	ID_S_T	int	Key field
2	Fk_ID_D	int	The secondary key is used to relate to the F table
3	Fk_ID_F	int	The secondary key is used to relate to the daemon (procedure)
4	Name_S	nvarchar (50)	The slot name
5	Memo_S	nvarchar (MAX)	The field keeps a comment
6	Name_T	nvarchar (50)	The terminal name
7	Value_T	nvarchar (50)	The value of the terminal
8	Memo_T	nvarchar (MAX)	The field keeps a comment

The machine representation of the set of Slot-Terminal pairs is represented by the rows in Table 14.

TABLE XIV. RECORDS IN THE TABLE *S\_T*

ID_S_T	Fk_ID_F	Memo_S	Name_T	Value_T	Memo_T
1	1	The cycle key (ID) of the request (V1)	Terminal 1: Cycle ID	7	int
2	1	Date and time of receipt (V1)	Terminal 2: Date-time	08 03 2026 09:00	Date time
3	1	Customer status ID (V1)	Terminal 3: Customer Status Identifier	3	int
4	2	Initialization date and time (V2)	Terminal 1 – Date and time of event triggering	08 03 2026 09:30	Date time
5	3	Initialization date and time (V3)	Terminal 1 – Date and time of event triggering	08 03 2026 10:00	Date time
6	4	Initialization date and time (V4)	Terminal 1 – Date and time of event triggering	08 03 2026 10:00	Date time
7	5	Initialization date and time (V5)	Terminal 1 – Date and time of event triggering	NULL	Date time
8	6	Initialization date and time (V6)	Terminal 1 – Date and time of event triggering	NULL	Date time
9	7	Initialization date and time (V7)	Terminal 1 – Date and time of event triggering	08 03 2026 10:30	Date time
10	8	Initialization date and time (V8)	Terminal 1 – Date and time of event triggering	08 03 2026 11:00	Date time
11	9	Initialization date and time (V9)	Terminal 1 – Date and time of event triggering	08 03 2026 11:00	Date time
12	10	Initialization date and time (V10)	Terminal 1 – Date and time of event triggering	08 03 2026 12:00	Date time
13	11	Initialization date and time (V11)	Terminal 1 – Date and time of event triggering	08 03 2026 12:15	Date time

Let's describe the relationships between the tables (Fig. 3). The table  $F$  is related to the table  $S_T$  by the one-to-many relationship. Thus, an unlimited number of slot-terminal pairs correspond to a single frame name. The table  $F$  is also connected to table  $E$  by a one-to-many relationship. This is used to attach the frame to the edges of the graph. Similarly, table  $F$  is related to table  $V$  by a one-to-many relationship. This is used to attach the frame to the vertices of the graph.

The table  $V$  is related to the table  $Type_V$  by the one-to-many relationship. This is used to divide the set  $V$  into classes. The  $V$  table is also linked to the  $CD$  table by a one-to-many relationship to describe the graph structure.

Table  $E$  is connected to table  $CD$  by a one-to-many relationship to describe connections in the graph.

#### IV. CONCLUSION

The proposed method of building information support is based on the use of a conceptual graph template. It allows IT designers to create conceptual models with minimal cost. The use of the template graph allows one to create a graph conceptual model that is mapped into a multidimensional logical data schema. The proposed method is oriented at developing a multidimensional database schema. The method is based on a sequence of steps. They are the following: the research of the subject area; the creation of process models; the design of a conceptual model; the design of a multidimensional database model. The multidimensional model supports the use of business intelligence methods and allows IT staff to use OLAP analysis.

#### REFERENCES

- [1] M. Fauler, *Enterprise Application Templates (Signature Series)*. Moscow:Williams, 2012, 544 p.
- [2] E.A. Grishenkov, "Building conceptual database models using the Enterprise Data Space Model template," in *Control systems and information technology*, vol. 3(41), 2010, pp. 81–87.
- [3] M. Minsky, "A framework for representing knowledge," *MIT AI Laboratory Memo*, vol. 306, June 1974.
- [4] D.A. Belov, P.V. Misevich, "Using multimedia frames to build remote monitoring and control systems," in *Proc. 8th Int. Conf. Intellectual Systems INTELS'2008, Moscow: Rusaki, 2008, pp. 168-170*.
- [5] A.E. Ermilov, P.V. Misevich, "Application of the frame apparatus in information and software for automated systems," in *Control systems and information technology*, vol. 1(63), 2016, pp. 34–37.
- [6] D.A. Belov, "Multimedia frame apparatus as a basis for constructing remote monitoring and control systems," in *Proc. 7th Int. Conf. The future of technical science*, N.Novgorod: NSTU, 2008, pp. 29-30.
- [7] D.A. Belov, V.P. Khranilov, "Application of multimedia frames for constructing intelligent sensors for monitoring systems," in *Proc. Int. Conf. Information systems and technologies IST-2009*, N.Novgorod: NSTU, 2009, pp. 205-206.
- [8] EPC function. Available: <https://lenkom40.ru/funktsiya-epc> (URL)
- [9] P.S. Kulyasov, P.V. Misevich, "Technology for mapping a modified EPC diagram into a relational model data structure," in *Scientific and Technical Bulletin of the Volga Region*, vol. 7, 2025, pp. 147-153.

#### Authors

**Valerii P. Khranilov** is with the Nizhny Novgorod State Technical University, Nizhny Novgorod, Russian Federation, Computer technologies in designing and production department (e-mail: [hranilov@nntu.ru](mailto:hranilov@nntu.ru)).

**Pavel V. Misevich** is with the Nizhny Novgorod State Technical University, Nizhny Novgorod, Russian Federation, Computational systems and technologies department (e-mail: [p\\_misevich@mail.ru](mailto:p_misevich@mail.ru)).

**Pavel S. Kulyasov** is with the Nizhny Novgorod State Technical University, Nizhny Novgorod, Russian Federation, Computational systems and technologies department (e-mail: [p.kulyasov@gmail.com](mailto:p.kulyasov@gmail.com)).

**Elena N. Pankratova** is with the Nizhny Novgorod State Technical University, Nizhny Novgorod, Russian Federation, Foreign languages department (e-mail: [keibusan@gmail.com](mailto:keibusan@gmail.com)).