Ontological Engineering of Interrelated Processes in Complex Cyber-Physical Systems

V. V. Borisov[1] and A. E. Misnik[2](✉)

[1] Branch of the National Research University "MPEI" in Smolensk, Smolensk, Russia
[2] Inter-State Educational Institution of Higher Education "Belarusian-Russian University", Mogilev, Belarus
anton@misnik.by

Abstract. The processes in complex cyber-physical system are always interrelated, which means different processes share the same information. Such sharing requires the robust approach to create, update and maintain the ontologies of processes and cyber-physical system as a whole in order to create well-organized data structure and to avoid duplication of information. The resulting complete ontology of cyber-physical system should be a convenient and flexible basis for modeling processes and ensuring the functioning of information-analytical and system processes. Metagraphs are considered as the basis for building an applied system ontology and extracting the ontologies of processes. A modification of the metagraph is proposed to include events and system methods in the ontology. This approach to the construction of a metagraph allows to include the process component in the ontological model of the system as an integral part of it, which allows to flexibly and with lower time expenses design, update and maintain process models based on segments of the general ontological model. As a basis for modeling, it is proposed to use a temporal neuro-fuzzy variety of Petri nets. An example of the implementation of a software-instrumental environment for ontological engineering and further construction of models of processes of a complex cyber-physical system is presented.

Keywords: Cyber-physical systems · Ontological engineering · Ontology · Metagraph · Temporal neuro-fuzzy Petri nets

1 Introduction

The organization of the system and information-analytical processes in modern complex cyber-physical systems is an urgent task, the solution of which is significantly hampered with an increase in information flows in such systems. Information-analytical processes include the processes of collecting, processing, generalizing, assessing and predicting the state of systems, developing reasonable management decisions and assessing their feasibility. Data sources can be both technical subsystems (temperature and pressure sensors, etc.) and operators of industrial and technological processes in the system. Often, specialists are simply not able to review all the events and phenomena occurring within the system and the ones affecting the system from the outside, as well as their

S. Kovalev et al. (Eds.): IITI 2022, LNNS 566, pp. 413–423, 2023.
https://doi.org/10.1007/978-3-031-19620-1_39

combinations. There is a steady growth in the volume of information of different quality coming from heterogeneous sources, which is necessary for the analysis and development of control decisions in the course of the system's performance of the tasks set, associated with both the improvement of sensor technologies and communication lines, which leads to an increase in the frequency of sensor polling and the emergence of the possibility obtaining data from new sources [4, 5].

The features of complex cyber-physical systems are:

- the complexity of the structure, multicomponent, the presence of functional subsystems that solve various target tasks, a large number of parameters that characterize the processes of the system;
- dynamic change in the structure and parameters of the system;
- incomplete information about the functioning and state of the system;
- variety of external and anthropogenic impacts on the system;
- the presence of complex nonlinear relationships between parameters;
- the complexity of making managerial decisions to improve the efficiency of the system;
- limited opportunities for experimental studies of the system and ongoing processes;
- impossibility of using a unified approach to the creation of models of system and information-analytical processes occurring in such systems.

Complex cyber-physical systems operate in conditions of uncertainty, as well as close interaction of the main system and information-analytical processes, there is an acute problem of increasing the efficiency of the design of information-analytical processes and system processes [11].

A possible solution is the control of complex cyber-physical systems and their processes based on simulation [8]. As a basis for modeling, it is proposed to use a temporal neuro-fuzzy variety of Petri nets, which have established themselves as a convenient, visual, and at the same time mathematically rigorous formalism for modeling and analyzing complex systems and their inherent processes. They allow to model processes, interaction protocols, and control processes with a sufficient degree of detail and visualization. Petri nets allow to naturally describe synchronization, parallelism, conflict and causation, as well as to visualize the structure and functioning of complex systems.

The processes in complex cyber-physical system are always interrelated, which means different processes share the same information. Such sharing requires the robust approach to create, update and maintain the ontologies of processes and cyber-physical system as a whole in order to create well-organized data structure and to avoid duplication of information.

2 Ontology

While working with complex cyber-physical systems, there is a tendency to integrate different-modal subsystems into a single system. It is obvious that the integration of multi-modal systems should be carried out in the presence of a single conceptual interface between them. An ontology is proposed as such an interface.

Ontology is a precise specification of a certain area, which includes a vocabulary of terms of the subject area and a set of relationships that describe how these terms relate to each other.

The names of concepts and relations of the ontology are selected based on the terminology corresponding to natural language realities in this area in order to simplify the process of developing an ontology for expert and knowledge engineer. Every developer of a domain ontology, whenever possible, uses the generally accepted terminology (mainly for concept names) and your own preference for relation names.

Ontology is required:

- for sharing by people or software agents a common understanding of the structure of information;
- for the possibility of reuse of knowledge in the subject area;
- to make domain assumptions explicit;
- to separate domain knowledge from operational knowledge;
- for the analysis of knowledge in the subject area.

The main characteristics of ontological knowledge models are:

- explicit separation of the ontology from the knowledge base, ensuring correspondence between them; when the ontology changes, all knowledge bases generated on its basis must be is automatically brought into the form corresponding to the modified model;
- declarative graph (semantic) representation of ontology and knowledge base, which allows developers describe the necessary knowledge base "from top to bottom", gradually moving from general concepts to detail, elements that clarify general concepts and essences, while a description itself should be holistic, and not "fragmented" into separate variables and entities;
- description of the ontology in a declarative language, which should be fairly simple and understandable, at the same time powerful, allowing one to describe arbitrary ontologies oriented and adapted to terminology and form that are understandable for developers of components of systems with knowledge base;
- automatic generation of editors of knowledge bases by ontology, which makes it possible for experts to form and maintain knowledge bases without intermediaries.

3 Ontological Engineering

Ontological engineering is a process of designing and developing ontologies that combines two main technologies for designing complex cyber-physical systems - object-oriented and structural analysis. It includes the selection of the main classes of entities in the description of real interacting processes, relations between these classes, as well as a set of properties that determine their change and behavior during interaction.

The goals of ontological engineering are to increase the level of integration of information necessary for making managerial decisions, to increase the efficiency of information retrieval, to provide the possibility of joint knowledge processing based on a single semantic description of the knowledge space [3].

The task of ontological engineering is to create formalized electronic knowledge models. The objectives of these models are business-specific and may include [14]:

- execution of simulation modeling of processes in order to optimize them;
- quick receipt of logical conclusions based on a large amount of information, in order to support decision-making;
- providing accessibility for the perception of users of large volumes of complexly structured information, the exchange of knowledge between people;
- solution of a number of technical problems, primarily in the field of information systems integration.

Applied ontologies describe concepts that depend on both the task ontology and the domain ontology [9, 10]. Ontological engineering implies:

- definition of classes of concepts in ontology;
- guidance of taxonomy on classes;
- development of structures of concepts and situations;
- definition of the properties of concepts and the values of these properties;
- procedures for outputting and transforming situations.

The domain ontology of complex cyber-physical system is very dynamic. To design such an ontology, appropriate tools are needed.

4 Metagraphs

Metagraphs are a generalization of the concepts of graph structures used in different areas. The metagraph contains elements of both digraphs and hypergraphs. The metagraph itself is built on the basis of a hierarchical graph.

A metagraph is a collection of directed set-to-set mappings. Its promising applications is the modeling of data relations, each of which is viewed as a mapping from a set of key elements to a set of content elements and the modeling of workflow tasks, each of which can be viewed as a mapping from a set of input documents to a set of output documents [1].

Metagraph - defined by three sets:

$$S = \{V, MV, E\}, \tag{1}$$

where V is the set of vertices of the metagraph, MV is the set of meta vertices, E is the set of edges. A meta vertex of a metagraph is a vertex that includes the set of vertices v, $v \in V$.

The metagraph vertex can be attributive, that is, an arbitrary number of attributes can be associated with it.

$$v_i = \{A\}, \ v_i \in V, \tag{2}$$

where v_i is the vertex of the metagraph; A - set of attributes

The edge of a metagraph is characterized by a source and destination vertex and directionality and connects two vertices, in contrast to the hyperedge of a hypergraph, which can span multiple vertices. An edge can connect both vertices and meta vertices of a metagraph.

Thus, there are fundamental differences between the metagraph and the hypergraph. The hypergraph was invented as a formalism that allows you to model the complex order of traversing the vertices of a graph. This approach, in particular, has found application in the design of electronic microcircuits. The main purpose of metagraphs is to model complex hierarchical objects and systems [2]. A schematic representation of the metagraph in Fig. 1.

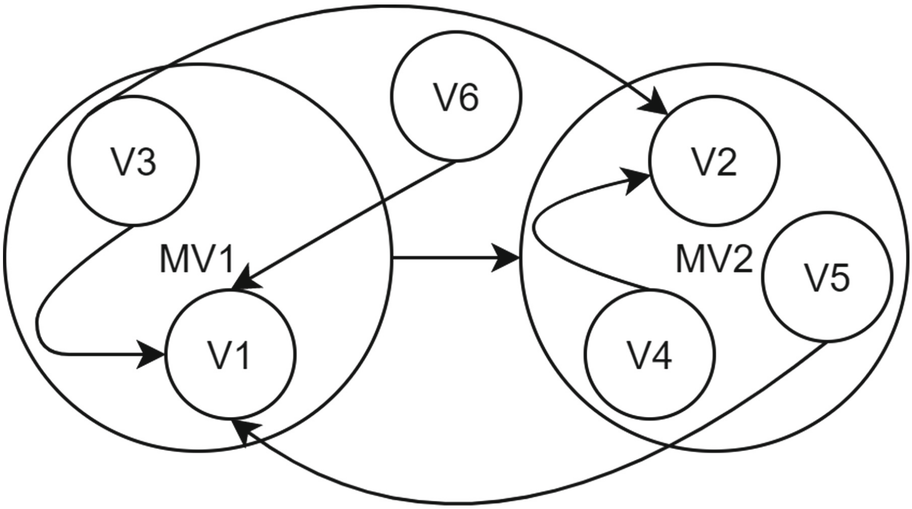

Fig. 1. Metagraph example.

It is proposed to generalize the concepts of a vertex and a meta vertex to the concept of a node of a metagraph (N) and to extend the definition of a node with sets of events and methods belonging to the node [12]:

$$N = \{N, A, EV, M\}, \tag{3}$$

where EV is the set of events associated with the metagraph node, M is the set of methods associated with the metagraph node.

Metagraph allows to define the ontology of the system. To describe the system data, meta vertices are used, which allow to define undirected semantic links between data elements. The presence of nesting, which is provided by meta vertexes, makes it possible to model complex hierarchical relationships between data elements.

This approach to building a metagraph allows to include the process component in the ontological model of the system as an integral part of it, which allows to design process models based on segments of the general ontological model in flexible and cost-effective way.

5 Temporal Neuro-Fuzzy Petri Nets

Ontology alone can be the basis of the model, but a technology is needed that provides the process of modeling, decision support or process control. We propose to use temporal neuro-fuzzy Petri nets as a tool for design of processes models, based on metagraph ontology representation [15].

The proposed temporal neuro-fuzzy Petri net (TNNSP) can be represented as follows:

$$TNFPN = (P, T, I, O, F, Al, D, m0, \alpha, R), \quad (4)$$

where $P = \{P_1, P_2, ..., P_{np}\}$ is a finite non-empty set of Petri net vertices, np is the total number of vertices; $T = \{T_1, T_2, ..., T_{nt}\}$ is a finite non-empty set of Petri net transitions, *nt* is the total number of transitions; $I = \{I_1, I_2, ..., I_{nt}\}$ – input transition functions; $O = \{O_1, O_2, ..., O_{nt}\}$ – output transition functions; $F = \{F_1, F_2, ..., F_{nt}\}$ is a finite non-empty set of the maximum execution time of Petri net transitions; $Al = \{Al_1, Al_2, ..., Al_{na}\}$ is the set of temporal Allen logic rules for network transitions, *na* is the total number of rules; $D = \{D_1, D_2, ..., D_{nd}\}$ is a finite set of label values required to trigger a transition when moving along an arc, *nd* is the total number of arcs; *m0* is the initial marking vector, each component of which is determined by the value of the membership function of the fuzzy presence of the marker in the corresponding position; $\alpha = \{\alpha_1, \alpha_2, ..., \alpha_{nt}\}$ is the vector of transition firing values, $R = \{R_1, R_2, ..., R_{nt}\}$ is a finite set of resources.

Figure 2 shows the generalized structure of the TNNSP.

The use of neuro-fuzzy transitions in Petri nets implemented using fuzzy neurons makes it possible to take into account uncertainty and previously obtained results of structural-parametric tuning [6, 7].

Modeling of processes can be characterized by various trajectories, the reflection of which is difficult for classical or neuro-fuzzy Petri networks, and the choice of the actual trajectory is an exhaustive task.

The use of Allen's temporal logic for the proposed model makes it possible to describe the sequences of simulated events, their interrelationships along the time scale and take into account temporal cause-and-effect relationships. A feature of Allen's temporal logic is the use of time intervals. It also allows to reserve and adaptively distribute resources at time intervals within the modeled process [13].

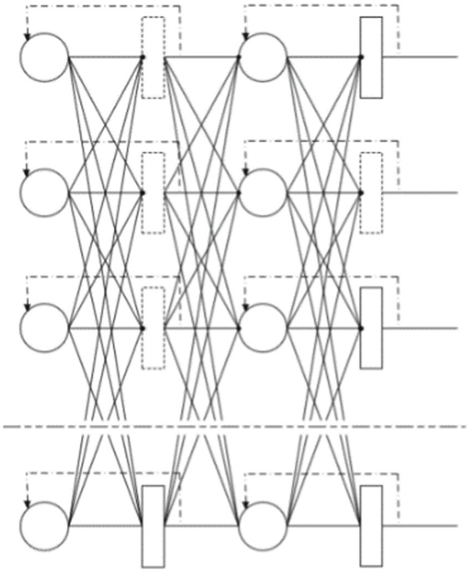

Fig. 2. The generalized structure of the temporal neuro-fuzzy Petri net.

As a result, it is possible to significantly reduce the number of possible trajectories for the implementation of the process. Figure 3 shows an example of the structure of the TNNSP, modeling the selected trajectory of the process, taking into account the restrictions imposed by temporary cause-and-effect relationships.

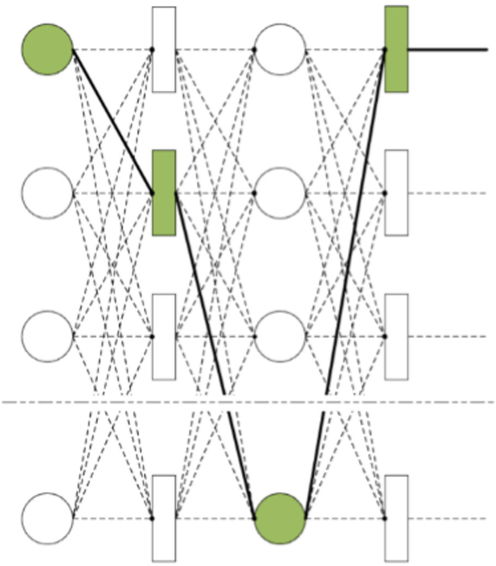

Fig. 3. An example of the structure of the TNGSP, modeling the trajectory of the process.

Figure 4 shows a fragment of the TNNSP, illustrating the joint use of neuro-fuzzy elements and temporal cause-and-effect relationships (implemented using the rules of Allen's temporal logic) between sequences of events when modeling an information technology process under uncertainty.

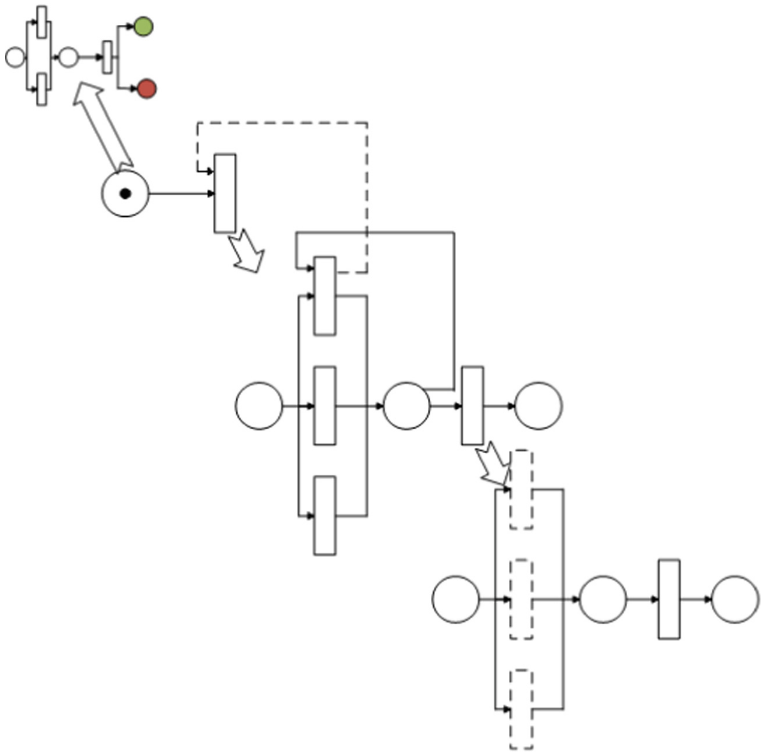

Fig. 4. Fragment of TNNSP illustrating the joint use of neuro-fuzzy elements and rules Allen's temporal logic.

6 Specialized Software

A specialized software and instrumental environment has been developed within which it is possible to model system processes and implement information and analytical processes of a complex cyber-physical system.

An important component of this environment is the ontological engineering module, which implements the principles of constructing an ontological model of a system as a metagraph with nodes that include events and methods (see Fig. 5).

This module allows to build the structure of a metagraph, set sets of attributes for its nodes, include many metagraph nodes in a node, including the possibility of recursive use of nodes, create methods for processing data and indicate the necessary events to activate them.

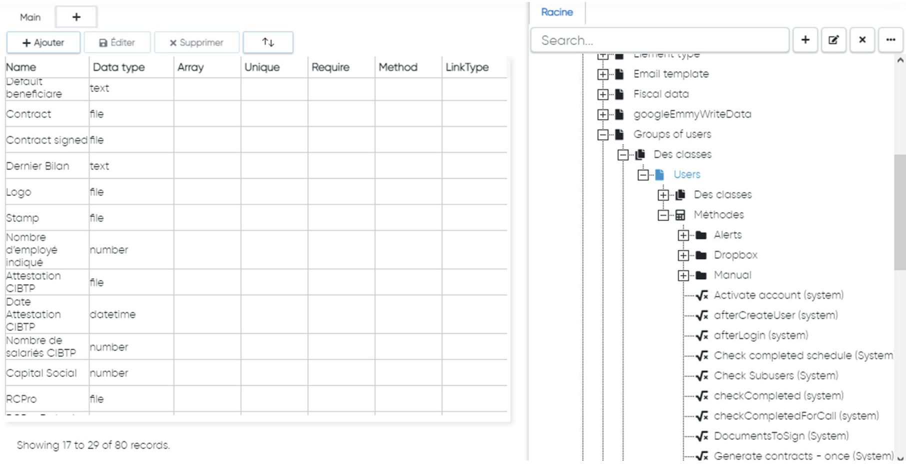

Fig. 5. Screen form of ontological engineering module.

Figure 6 shows the screen form of processes design module based on temporal neuro-fuzzy Petri nets.

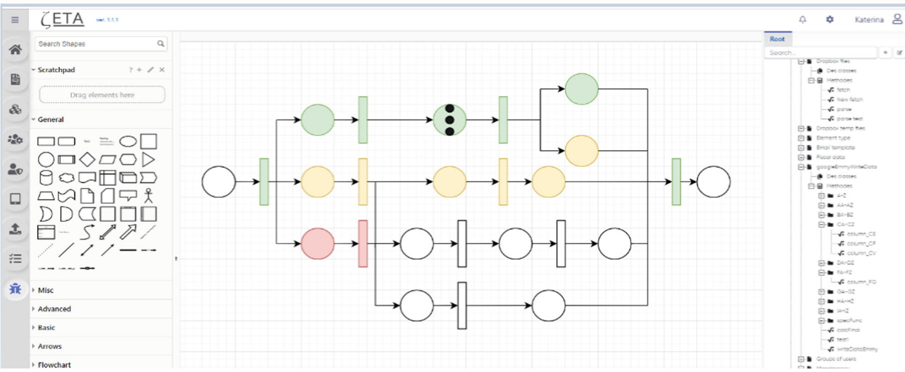

Fig. 6. Screen form of process modeling module based on temporal neuro-fuzzy Petri nets.

7 Conclusion

The universal means of representing of hierarchy, attributes, events, and methods within the ontology are required to effectively solve the problem of ontological engineering.

The tools for ontological engineering of interrelated processes are considered. The use of metagraphs as an adequate and convenient tool for ontological engineering is justified.

The modification of metagraphs for ontology design of a complex cyber-physical system and its interrelated processes is proposed.

The modification of temporal neuro-fuzzy Petri net for interrelated process modeling is proposed.

An example of the implementation of a software-instrumental environment for ontological engineering and further construction of models of processes of a complex cyber-physical system is presented.

Acknowledgements. This study was supported by the Russian Science Foundation (project no. 22-61-00096).

References

1. Basu, A., Blanning, R.: Metagraphs and their Applications, 174 p. Springer, Heidelberg (2007). https://doi.org/10.1007/978-0-387-37234-1
2. Samokhvalov, E.N., Revunkov, G.I., Gapaniuk, Iu.E.: Ispol'zovanie metagrafov dlia opisaniia semantiki i pragmatiki informatsionnykh sistem. Vestnik MGTU im. N.E. Baumana, ser. "Priborostroenie", vyp. no. 1, pp. 83–99 (2015)
3. Gribova, V., Shalfeeva, E.: Ontological infrastructures for intelligent problem solving. In: CEUR Workshop Proceedings: Russian Advances in Fuzzy Systems and Soft Computing: Selected Contributions to the 10th International Conference on «Integrated Models and Soft Computing in Artificial Intelligence», IMSC 2021, vol. 2965 (2021)
4. Bobryakov, A.V.: A method for managing engineers training processes using swarm intelligence algorithms. In: Bobryakov, A.V., Yanukovich, S.P., Zakharchenkov, K.V., Borisov, V.V. (eds.) 2020 5th International Conference on Information Technologies in Engineering Education, Inforino 2020 – Moscow, Russia (2020). https://doi.org/10.1109/Inforino48376.2020.9111788
5. Bobryakov, A., Misnik, A., Prakapenka, S.: Management of industrial and technological processes of complex systems based on modified neuro-fuzzy Petri nets. In: CEUR Workshop Proceedings: Russian Advances in Fuzzy Systems and Soft Computing: Selected Contributions to the 10th International Conference on «Integrated Models and Soft Computing in Artificial Intelligence», IMSC 2021, vol. 2965, pp. 276–283 (2021)
6. Borisov, V.V., Zakharchenkov, K.V., Kutuzov, V.V., Misnik, A.E., Prokopenko, S.A.: Modeling educational processes based on neuro-fuzzy temporal Petri nets. Appl. Inform. **16**(4), 35–47 (2021). https://doi.org/10.37791/2687-0649-2021-16-4-35-47
7. Bobryakov, A.V., Krutalevich, S.K., Misnik, A.E., Prokopenko, S.A.: Modeling of industrial and technological processes in complex systems based on neurofuzzy petri nets. Journal of Physics: Conference Series, Volume 2096, International Conference on Automatics and Energy (ICAE 2021), 7–8 October 2021, Vladivostok, Russia (2021)
8. Vajnilovich, Yu., Zaharchenkov, K., Zakharova, A.: Integrated system approach to improving the efficiency of IT projects management based on evolutionary modeling. Computing in Physics and Technology, pp. 309–314
9. Gomez-Perez, A., Fernández-López, M., Corcho, O.: Ontological Engineering: With Examples from the Areas of Knowledge Management, E-Commerce and the Semantic Web (2004)
10. Arpirez, J., Gomez-Perez, A., Lozano, A., Pinto, S.: 2Agent: an ontology- based WWW broker to select Ontologies. In: Workshop on Applications of ontologies and Problem Solving Methods, ECAI 1998 (1998)
11. Soshnikov, D.: Software toolkit for building embedded and distributed knowledge-based systems. In: Proceedings of the 2nd International Workshop on Computer Science and Information Technologies, CSIT 2000. Ufa, pp. 103–111 (2000)

12. Misnik, A.E.: Ontological engineering on metagraphs basis. In: 2022 VI International Conference on Information Technologies in Engineering Education (Inforino), pp. 1–6 (2022). https://doi.org/10.1109/Inforino53888.2022.9782909
13. Bobryakov, A.V., Borisov, V.V., Misnik, A.E., Prokopenko, S.A.: Design and implementation of information-analytical and industrial and technological processes in production based on neuro-fuzzy Petri Nets. In: 2022 VI International Conference on Information Technologies in Engineering Education (Inforino), pp. 1–6 (2022). https://doi.org/10.1109/Inforino53888.2022.9782997
14. Bobryakov, A.V., Borisov, V.V., Gavrilov, A.I., Tikhonova, E.A.: Compositional fuzzy modeling of energy- and resource saving in socio-technical systems. EAI Endorsed Trans. Energy Web Inf. Technol. **6**, e9 (2019). https://doi.org/10.4108/eai.12-9-2018.155863
15. Borisov, V., Luferov, V.: Method for forecasting multidimensional time series based on neuro-fuzzy cognitive temporal models. In: Kovalev, S., Tarassov, V., Snasel, V., Sukhanov, A. (eds.) IITI 2021. LNNS, vol. 330, pp. 149–160. Springer, Cham (2022). https://doi.org/10.1007/978-3-030-87178-9_15