

Development of an Approach to Building Representation Metaphors for Visualizing Graph Models (by the Example of Cognitive Maps)

Ruslan Isaev¹ and Aleksandr Podvesovskii¹

¹ Bryansk State Technical University, 7, 50 let Oktyabrya blvd., Bryansk, 241035, Russia

Abstract

The paper presents development of the authors' approach to visualization of graph models of various types based on the use of visualization metaphors and aimed at increasing cognitive clarity of these models. One of the key problems of this approach is investigated – namely, formalization of the process of constructing representation metaphors for graph models. Features of graph models that allow formalizing the process of their visualization are considered, the necessary terminology is introduced. A number of principles have been formulated that must be considered when forming metaphors for representing graph models. On the basis of the introduced principles, a general approach to the construction of representation metaphors for visualization of arbitrary graph models is proposed. The main ideas for applying the proposed approach are demonstrated by the example of a fuzzy cognitive map when constructing a metaphor for representing the results of its structure and target analysis. The directions of advanced research have been determined, within the framework of which it is planned to further formalize the approach in order to ensure the possibility of its software implementation. In the future, the presented approach can become an important component of an integrated approach to building a visualization mechanism for an arbitrary graph model, which provides support for efficient visual analysis throughout all stages of modeling.

Keywords

Graph model, graph visualization, visualization metaphor, cognitive map.

1. Introduction

In modern theory and practice of knowledge engineering and decision making, it is often required to deal with models that can be represented in the form of graphs. Examples of graph models include the following:

- Semantic networks, thesauri, ontologies [1];
- Bayesian networks, as well as influence diagrams based on them [2, 3];
- Decision trees [4], probabilistic decision trees [3];
- Markov decision process models [2, 5];
- Models of analytic hierarchy process and analytic network process [6];
- Transshipment models [5];
- Cognitive models based on different types of cognitive maps [7].

It is characteristic that it is the graph form of representation of both the listed and other similar models that is usually the most natural and intuitive for user perception. Indeed, each of these models can be easily associated with a graph with vertices corresponding to the main elements of the object, system or situation under consideration (for example, these can be elements of a decision-making problem or a model of knowledge about a certain subject area), and the edges between the vertices

GraphiCon 2021: 31st International Conference on Computer Graphics and Vision, September 27-30, 2021, Nizhny Novgorod, Russia

EMAIL: ruslan-isaev-32@yandex.ru (R. Isaev); apodv@tu-bryansk.ru (A. Podvesovskii)

ORCID: 0000-0003-3263-4051 (R. Isaev); 0000-0002-1118-3266 (A. Podvesovskii)



© 2021 Copyright for this paper by its authors.

Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

correspond to relations between the respective elements. Depending on the type of a model, the vertices of a graph can be either homogeneous (i.e., represent “equal” elements of the same nature) or heterogeneous (for example, there are qualitatively different types of nodes in decision trees). A similar statement is true for the edges of a graph. Semantic interpretations of vertices and edges determine characteristics of a graph. So, it can be directed or undirected, be or not be weighted, allow cycles or be acyclic, etc. Also, a mathematical apparatus, used within the framework of a specific type of graph model, plays an important role: for example, probabilistic models [2, 3], fuzzy models [7, 8], etc. can be distinguished.

The presence in the discussed models of the graph form of representation naturally leads to the problem of their visualization. It is characterized by the availability of many possible ways to solve it, among which, as a rule, there is no predominantly correct way. Taking this into account, an approach can be used to describe the visualization problem in general which is based on the concept of a visualization metaphor [9]. A visualization metaphor is understood as a set of principles for transferring characteristics of the object under study into the space of a visual model. The visualization metaphor includes two components, applied sequentially:

- a *spatial metaphor*, describing the general principles of building a visual model (in particular, type and dimension of visualization space, relative position of model elements);
- a *representation metaphor*, which is responsible for clarifying characteristics of a visual image (as a rule, with the aim of visualizing certain properties of the object under study which are of most significance at the current stage of its analysis).

An important aspect of working with any graph model that affects the efficiency of its application is the simplicity of perception of the model by the researcher. To describe this aspect, a concept of cognitive clarity is often used [10], which means the ease of intuitive understanding and interpretation of a certain amount of information presented in a certain form. Lack of cognitive clarity is usually associated with difficulty in understanding information, with missing a significant part of it, inaccurate or erroneous interpretation of some of its elements, etc. As applied to a graph model, ensuring a high level of cognitive clarity of its representation allows the researcher to notice more important properties of the model “at a glance”, to find more errors made in its construction, and also to interpret the results of its analysis faster.

2. Visualization metaphors as a means of increasing cognitive clarity of graph models

In [11-14], the authors investigated aspects of application of visualization metaphors in the problems of visualization of fuzzy cognitive maps and graph models in general. Summarizing the results obtained, the process of visualizing a graph model using a visualization metaphor can be represented as a diagram in Fig. 1. Thus, a spatial metaphor determines the arrangement principle of graph vertices and edges in a visual space, therefore, it can be based on well-known graph tiling algorithms. Following [15], we say that the result of applying a spatial metaphor is a *spatial arrangement* of a graph model. In turn, a representation metaphor is intended to focus the researcher’s attention on certain aspects or results of modeling depending on his needs at a particular stage of working with the model. For this, visual features of graph vertices and edges can be used displaying the attribute values of the corresponding model elements in a cognitively accessible form. This is how a ***visual representation*** of a graph model is formed. Spatial arrangement and visual representation together form a *visual image* of a graph model.

In [14], the authors, using an example of visualization of fuzzy cognitive maps, showed that use of visualization metaphors allows structuring and partially formalizing the task of increasing cognitive clarity of a visual representation. They also hypothesized that a similar effect can be achieved through the use of visualization metaphors for graph models of other types.

Accordingly, each of constituents of a visualization metaphor must contribute to enhancing cognitive clarity of a graph model. At the same time, the spatial metaphor provides an increase in cognitive clarity mainly due to optimization of a number of formal indicators characterizing graph tiling, which are taken as criteria for cognitive clarity at this stage. The authors proposed [11] and formalized [12] a basic set of criteria that can be taken as a basis for assessing cognitive clarity of any type of graph models. Also, in [12], an approach was proposed that allows automating comparison of

a set of generated tilings of a given graph in order to select the one that provides the greatest cognitive clarity of a visual image.

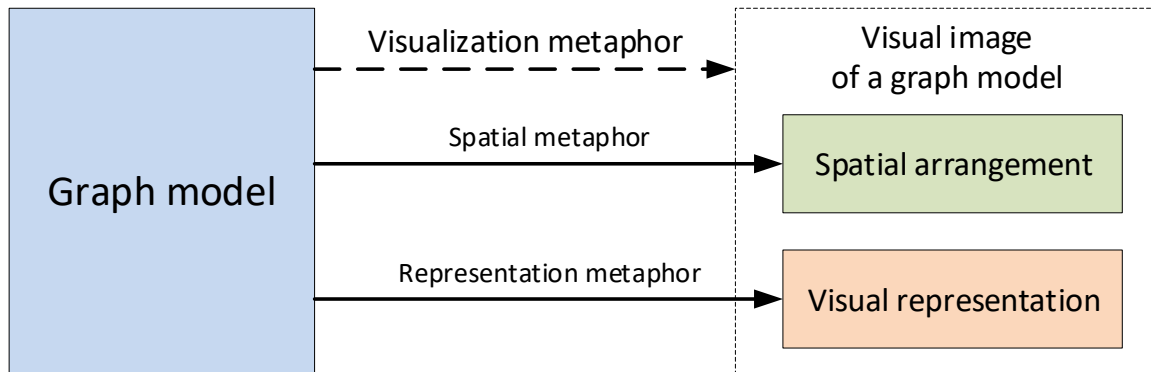


Figure 1: Graph model visualization using a metaphor

Thus, at the moment, the task of constructing spatial metaphors of graph models has a general solution, which is later to be elaborated and adapted for various types of models considering their specific features. At the same time, we are also interested in formalization and automation of the process of constructing qualitative metaphors for representing graph models which provide a high level of cognitive clarity of these models in their visual analysis. Next, we will consider some ways to solve this problem.

3. Grounds for formalizing the process of constructing representation metaphors for graph models

It is necessary to highlight a number of characteristic features inherent in graph models that are significant in the context of their visualization.

First, each specific type of graph models is characterized by a certain structure which can be formally described. So, in the general case, the main elements of the model (i.e., graph vertices) can belong to one of several types, the conceptual meaning and internal structure of which, as a rule, are specified and described in advance. The same applies to relations between elements (graph edges) – the acceptable types of such relations and the corresponding conceptual interpretations are usually known in advance. Further, to simplify the terminology used, we will call structural components of the corresponding graph – both vertices and edges – model elements.

Secondly, as a rule, model elements are assigned some attributes (i.e. properties, characteristics of these elements) that have certain tolerance ranges and interpretations and can also have an internal structure, i.e. contain a number of simpler attributes. Elementary attributes that have no internal structure, from the point of view of tolerance ranges, usually correspond to elementary data types: text strings, integers or real numbers (often from certain ranges), elements of discrete sets, binary yes/no values, etc. In fact, attributes make up the parametric space of a graph model and can reflect both its initial data (that is, specified when building the model) and the results of its analysis.

Thirdly, quite an obvious solution is to visualize elements and attributes of different types by different methods. Meanwhile, for each specific type, it is possible to distinguish (both intuitively and on the basis of experience) visualization methods that are more preferable from the point of view of ensuring a high level of cognitive clarity.

Let us say that a model element is visualized by creating a *visual image* of this element, and an attribute of the element is rendered by assigning some *visual feature* corresponding to the visual image. The graph model as a whole is visualized by creating a *visual representation*, i.e. a set of visual images of model elements, the visual features of which reflect the attributes of these elements.

With this consideration in mind, the above visualization method can be more formally understood as establishing a correspondence between a specific type of an element (attribute) and a specific type of a visual image (visual feature), and a representation metaphor as a whole – as a necessary set of such visualization methods.

When choosing methods to visualize attributes, it is advisable to proceed from their characteristics. Thus, a type of tolerance range of the attribute must be considered, in particular, whether it is discrete or continuous. It should also be taken into account whether it is important to know the exact value of the attribute from the point of view of the visual analysis of the model, or whether some “approximate picture” which provides a qualitative insight into the situation is sufficient. Otherwise, the choice of visualization methods is predominantly subjective. In addition, in some cases it can be dictated by already established traditions. Such a situation often arises in cases of widespread use of software tools for supporting a certain type of model (for example, it is characteristic of Bayesian networks). In any case, possible ways to visualize various types of attributes can be represented in a certain formalized form, thus forming a knowledge base suitable for use in developing representation metaphors for any graph models.

Also, when building representation metaphors, it is important to consider that various attributes of model elements become relevant at different stages of modeling. For example, at the stage of building a model, attributes representing the results of its analysis will be irrelevant (since at this stage they do not yet have definite values). In addition, if the process of model analysis includes a number of logically separate stages (as, for example, in the case of cognitive maps where it is customary to distinguish structure and target and scenario stages of analysis), then such a model is characterized by the presence of several separate groups of element attributes. In turn, each of the stages can be logically divided into sub-stages, which leads to emergence of subgroups of attributes, etc. All this allows us to introduce the concept of a graph model *representation*, by which we mean a special relation between elements and their attributes. The representation selects from the set of all element attributes a subset of those that are to be visualized. Thus, when constructing metaphors for visualizing graph models, the representation can be used as a named template that makes it easier for the analyst to select model elements and attributes for solving a specific visual analysis problem.

The material presented in this section creates the basis for formalization and partial automation of the process of developing representation metaphors for graph models.

4. Principles of forming representation metaphors for graph models

Let us formulate a number of principles that determine the rules of forming representation metaphors for graph models. These principles should be considered when developing approaches to constructing such metaphors. The semantic content of these principles is schematically illustrated in Fig. 2.

1. *The principle of partial visualization.* As a rule, only some subset of elements and their attributes available in the model are visualized “at one point in time” (or, based on the terminology introduced above, one representation is visualized). This is due both to the high structural and parametric complexity of graph models, which, as a rule, exceeds the analyst’s cognitive capabilities, and to the multi-stage process of studying models, when at a certain stage there is a possibility and need to visualize only a part of the information related to the model.
2. *The principle of injective visualization.* Different attributes of model elements within the same representation metaphor must be visualized in different ways. In other words, mixing of two or more attributes within one visual feature is not allowed, since this will entail mixing of the corresponding properties of the model in the analyst’s perception.
3. *The principle of surjective visualization.* Each separate visual feature must reflect a specific attribute that is significant in the context of the problem being solved. In other words, the researcher’s perception of the model must not be cluttered with information that is irrelevant in the context of the problem, since this can lead to a slowdown in visual analysis.
4. *The principle of subordination.* Each subordinate element of the model must be visualized in such a way that its visual image makes it possible to unambiguously establish which particular element it is subordinate to. A special case of subordination is logical nesting of one element of the model in another, which should be displayed, respectively, as nesting of visual images.
5. *The principle of restructuring.* In some cases, it is possible to merge two discrete attributes into one attribute based on the Cartesian product of their tolerance ranges. So, in the example considered below (Table 1), it is allowed to merge the “Type” and “Target” attributes (the resulting attribute takes on the values “unmanaged target”, etc.). The reverse variant of application of this principle is

also possible: the original attribute is split into two attributes of different types. For example, the attribute “Influence magnitude” can be divided into “Influence sign” and “Influence intensity” (Table 2). In general, application of this principle allows optimizing a representation metaphor due to a more rational use of the space of available visual features.

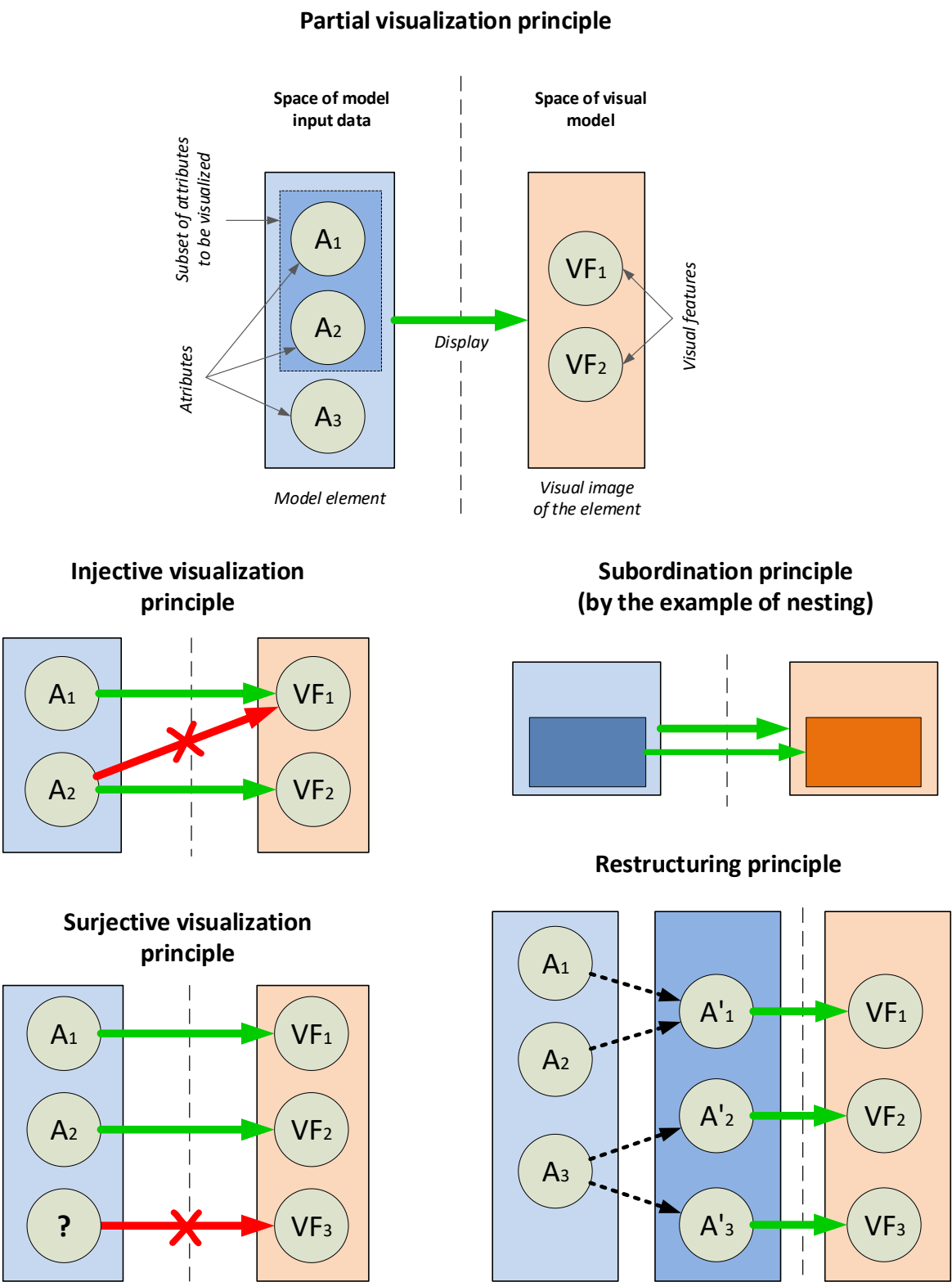


Figure 2: Illustration of the principles of forming representation metaphors for graph models

5. General approach to construction of representation metaphors for graph models

Considering the formulated principles, it is possible to propose a general approach to construction of representation metaphors for graph models. It can be schematically represented as a generalized algorithm (Fig. 3). Let us briefly describe its main stages.

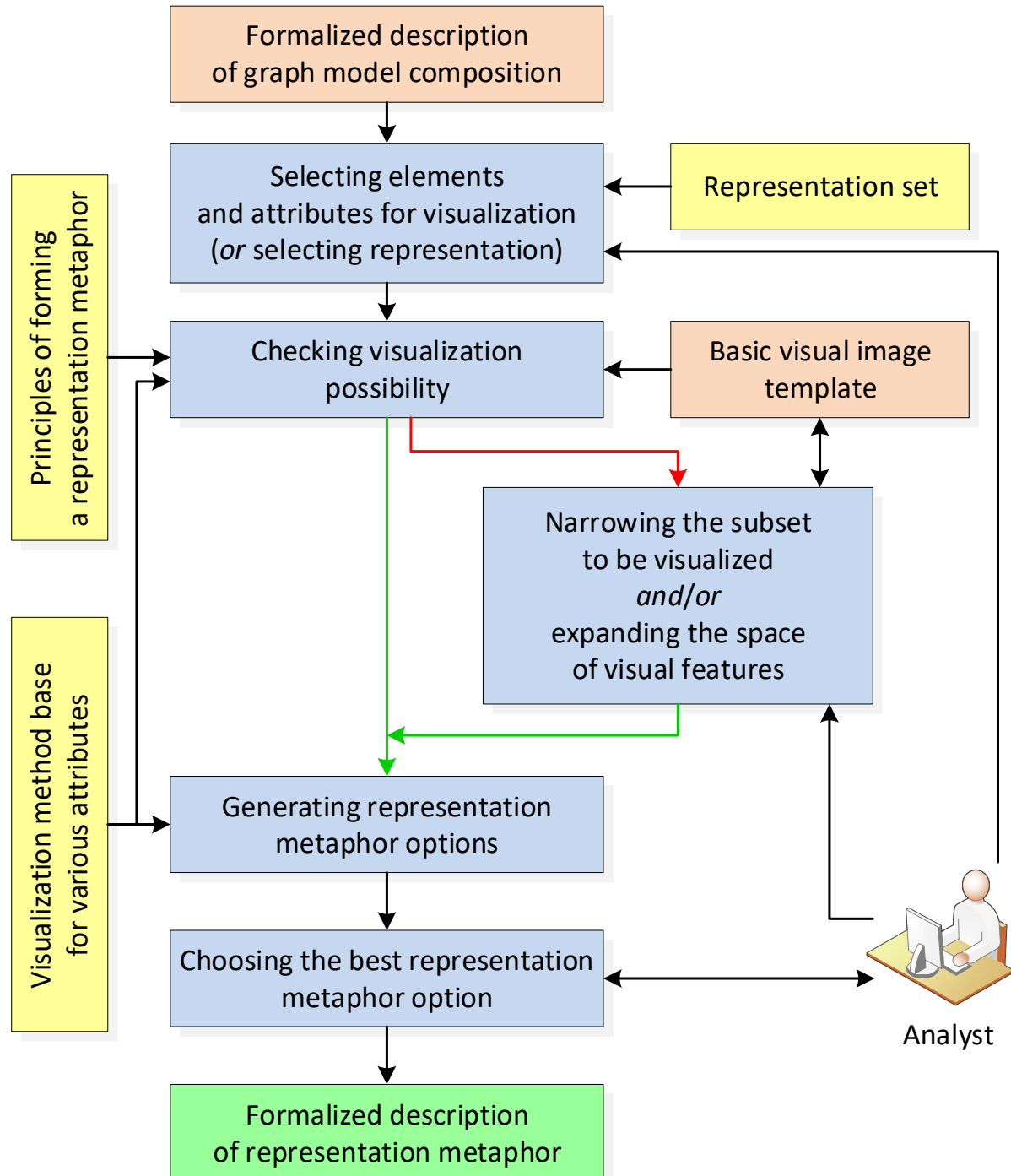


Figure3: Generalized algorithm for constructing representation metaphors for graph models

1. The analyst chooses a subset from the set of elements and attributes of the model to be visualized within one representation metaphor. This stage can be performed in an accelerated version, by choosing a specific representation from a set of possible representations (if there is a

similar set which can be formed, for example, on the basis of previous experience with graph models of this type).

2. Checking the possibility of simultaneous visualization of all attributes from the selected subset. In fact, at this stage, an attempt is made to establish a correspondence between the specified attributes and the available visual features taking into account the known methods of visualizing attributes of various types. Available visual features are formalized by means of a basic template of a visual image – a pre-compiled structure that stores a hierarchy of visual features with indications of their types. At this stage, compliance with the principles of injective and surjective visualization as well as the principle of subordination is ensured. If necessary, it is possible to use the restructuring principle, while the decision to merge or split attributes is made by the analyst.

3. If it is impossible to establish at least one correspondence option, the analyst can be offered to narrow down the subset of visualized attributes. The excluded attributes can be visualized in the future using a different representation metaphor. Another way to achieve the desired correspondence is to expand the space of available visual features by modifying the visual image template, which is also performed by the analyst. In each of the two methods, it is possible and advisable to formulate recommendations for the analyst on the choice of an optimal course of action to achieve the required result.

4. Generating possible options for representation metaphor implementation, which is carried out according to the principle of enumerating visualization methods (i.e., combining acceptable correspondences between attributes and visual features), considering the given preference of these methods.

5. Analyst's visual familiarization with the obtained options of the representation metaphor and selection of the most preferable one (performed considering subjective informal preferences).

6. An example of constructing a representation metaphor for a fuzzy cognitive map

Let us consider an example of applying the proposed approach to constructing a representation metaphor for V.B. Sylov's fuzzy cognitive map (FCM) [16] within the framework of a cognitive model of analysis and planning of software projects [17]. In the example, the details related to formalization of data representation and processing will be omitted; the emphasis will be on the key features of the proposed approach.

Elements of a cognitive map are concepts of the studied subject area (software project management) and have a single set of attributes, thus being "homogeneous", i.e. belonging to the same type. All relations between elements (relations between concepts) existing in the model define cause-and-effect influences between them, have common sets of attributes and, thus, also belong to the same type.

Tables 1 and 2 show concept attributes and cognitive model influences (in the case of concepts, a small sample of attributes is given), for which tolerance ranges and examples of visualization methods that provide sufficient cognitive clarity are indicated. When listing the attributes, the authors relied on the implementation of Sylov's FCM apparatus within the framework of IGLA decision support system [18].

Suppose the researcher performs the stage of structure and target analysis of the cognitive model and is going to visualize the following concept attributes at the same time: name, type, influence on the system, as well as all available attributes of relations between concepts. Meanwhile, suppose that the basic template of the concept visual image contains only two visual features: the text displayed on it and the background color (which corresponds to the quest for creating the simplest metaphors with high cognitive clarity).

At the second stage of the algorithm, it was found that it was impossible to establish a correspondence between attributes and visual features under the indicated conditions. According to known visualization methods (which could be obtained by formalizing knowledge from Table 1), the displayed text is matched to the concept name, but the background color of the concept cannot simultaneously reflect its two other attributes (type and influence on the system). Accordingly, a violation of the principle of injective visualization has occurred.

Table 1

Some attributes of a concept within Sylov's FCM and their visualization methods

Attribute	Attribute tolerance range	Examples of visualization methods
Name	Text strings	Displaying text (probably abbreviated)
Type	Discrete set, for example: {unmanaged; managed; observable}	- Discrete color coding with contrasting colors - Different shape of vertices
Target	Yes/No	- Binary color coding - Presence/absence of a binary visual feature (frame, icon, ...) - Different shape of vertices
Influence on the system	Real numbers from the range $[-1; 1]$	- Continuous color coding with rendering the influence sign through the color and the influence strength through the color intensity - Bar graph

Table 2

Attributes of influence within Sylov's FCM and their visualization methods

Attribute	Attribute tolerance range	Examples of visualization methods
Concept of cause	Set of concepts	The relation comes visually from the vertex corresponding to the concept
Concept of effect	Set of concepts	The relation visually enters the vertex corresponding to the concept, which is indicated by a marker (usually an arrow)
Influence sign	Positive/ negative	- Binary color coding - Various line styles (solid, dashed, etc.)
Influence intensity	Real numbers from the range $(0; 1]$	- Line thickness - Color intensity

The representation metaphor for the cognitive model shown in Fig. 4 can be formed after narrowing the set of attributes to be visualized: the analyst agrees to visualize only the name of the concept and its influence on the system. In contrast, the metaphor in Fig. 5 can be obtained as a result of adding to the visual image template a new graphic element which provides the missing visual feature.

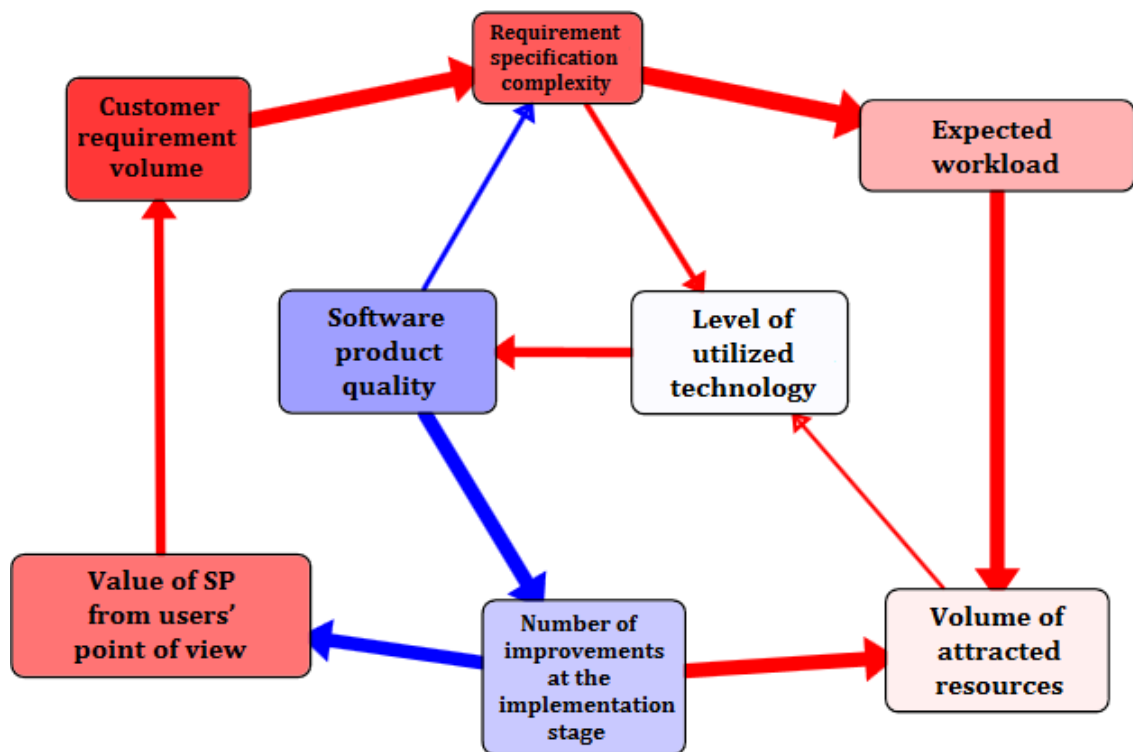


Figure 4: An example of an FCM representation metaphor

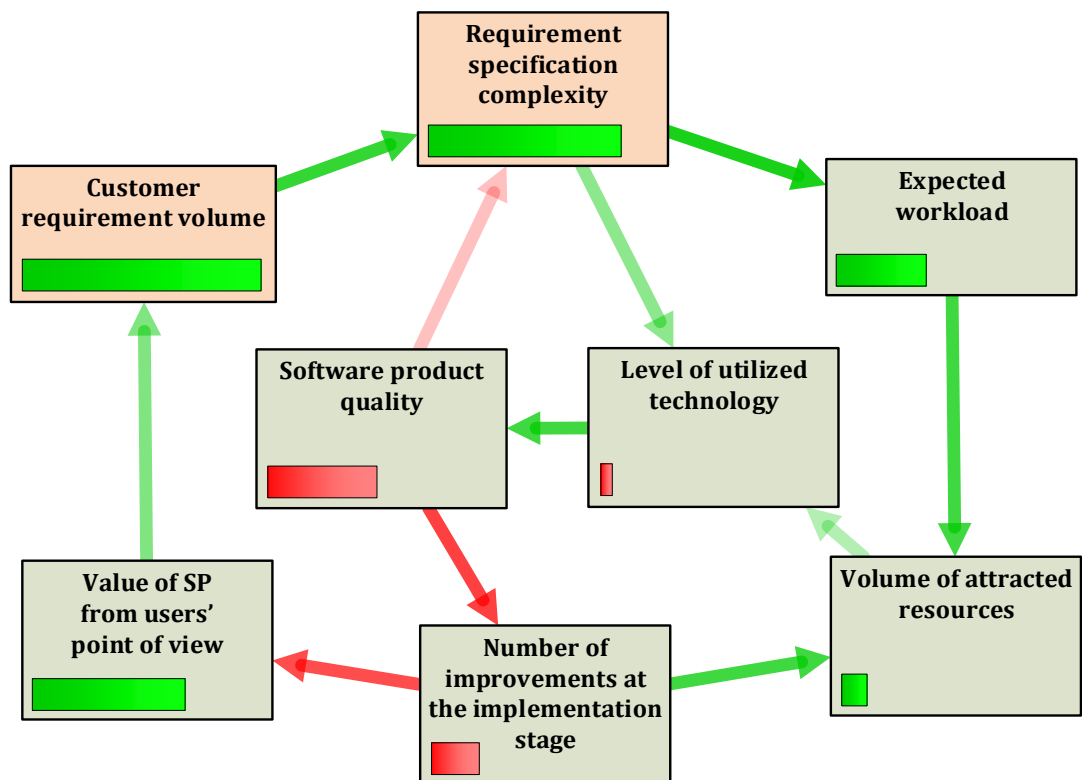


Figure 5: An example of an FCM representation metaphor: alternative selection of a subset of attributes and methods of their visualization

The key difference between these representation metaphors is the way of visualizing the influences of concepts on the system: the colors of the graph vertices (Fig. 4) and the elements of the bar graph distributed over the set of vertices (Fig. 5) are used as visual features. This allowed, in the second metaphor, to “release” the vertex color for visualization of the concept type (thus, in this cognitive model, the concepts “Customer requirement volume” and “Requirement specification complexity” are managed, i.e. direct control actions can be exerted on the corresponding parameters of the system, the other concepts are unmanaged). Thus, the advantage of the second metaphor is the simultaneous visualization of the concept type and its influence on the system. Due to this, the analyst can obtain a larger amount of information of interest to him in the course of one act of visual perception of the cognitive model. The “trade-off” for this is the speed of the very act of perception, which slows down due to the complication of the visual image, which, however, in the given example is insignificant.

The rest of the differences between the two metaphors (in terms of visualizing relations between concepts) can demonstrate flexibility of the proposed approach and consideration of the subjective component within its framework: the choice of the final version of the representation metaphor from a number of acceptable ones as well as adjustment of preferable color schemes remain with the analyst. Thus, in the above examples, different color schemes were used to visualize the signs of influences: red/blue (Fig. 4) or green/red (Fig. 5) to convey positive and negative influences, respectively. The influence intensity of one concept on another is transmitted through visual features of the thickness of the edge between them (Fig. 4) or the intensity of its color (Fig. 5).

7. Conclusion

The proposed approach can become one of the key components of an integrated approach to building a visualization mechanism for arbitrary graph models. It is assumed that this mechanism should be based on a system of visualization metaphors that provide an increase in cognitive clarity of a graph model throughout all stages of its construction and analysis.

At the same time, an interesting promising opportunity is to provide intelligent switching between visualization metaphors reflecting the optimal order in each specific case for changing the stages of modeling.

It appears that a certain paragon and a criterion for the success of application of the visualization mechanism can be the performance of all necessary actions with the graph model predominantly (or even exclusively) in a visual form. In other words, the visualization mechanism can be considered the more successfully constructed, the more work can be done with its use without involving alternative means and forms of information display.

A number of specific directions can be also indicated for the development of the proposed approach in the near future. They are focused primarily on ensuring the possibility of software implementation of the algorithm that underlies it:

- Formalization of the terminological apparatus introduced within the framework of the approach, in particular, the concepts of a graph model element, an attribute, a representation, a visual image and a visual feature.
- Development of a language (languages) for a formal description of the composition of the graph model and the structure of its visual image or adaptation to this task of any of the existing markup languages (for example, XML or JSON).
- Providing automated generation of a set of graph model representations based on a formal description of its composition.
- Development of a language for formal description of representation metaphors for graph models resulting from the application of this approach.

In the context of the indicated directions, a software tool for supporting visual analysis of various graph models of knowledge representation and decision making can become a promising applied result of the research carried out by the authors. This tool can be implemented in the form of a library for building visual analysis mechanisms, which can be used in the development or modernization of decision support systems and other software systems the work of which is closely related to information representation in the form of graphs.

8. Acknowledgements

The reported study was funded by RFBR, project number 19-07-00844.

9. References

- [1] S. Staab, R. Studer (Eds.), *Handbook on Ontologies*. Springer-Verlag Berlin Heidelberg, 2009. doi: 10.1007/978-3-540-92673-3
- [2] L.E. Sucar, *Probabilistic Graphical Models. Principles and Applications*. Springer-Verlag London:, 2015. doi: 10.1007/978-1-4471-6699-3.
- [3] F.V. Jensen, T.D. Nielsen, *Bayesian Networks and Decision Graphs*, 2nd. ed. Springer Science + Business Media LLC, 2007.
- [4] M. Bramer, *Principles of Data Mining*. Springer-Verlag London Ltd., 2016. doi: 10.1007/978-1-4471-7307-6
- [5] H.A. Taha, *Operations Research: An Introduction*, 10th. ed. Pearson, 2017.
- [6] T.L. Saaty, *Decision Making with Dependence and Feedback: The Analytic Network Process*. RWS Publishing, Pittsburgh, PA, 2001.
- [7] V.V. Borisov, V.V. Kruglov, A.S. Fedulov, *Fuzzy Models and Networks*. Hot Line – Telecom, Moscow, 2012 (in Russian).
- [8] A. Piegat, *Fuzzy Modeling and Control*. Physica-Verlag Heidelberg, 2001. doi: 10.1007/978-3-7908-1824-6
- [9] A.A. Zakharova, A.V. Shklyar, *Visualization Metaphors*. *Scientific Visualization* 5 (2) 16-24 (2013).
- [10] W. Huang, S.H. Hong, P. Eades, *Predicting Graph Reading Performance: A Cognitive Approach*. in: *Proc. Asia Pacific Symposium on Information Visualization (APVIS2006)*, Tokyo, Japan, 2006, pp. 207–216. doi: 10.1145/1151903.1151933
- [11] A.G. Podvesovskii, R.A. Isaev, *Visualization Metaphors for Fuzzy Cognitive Maps*. *Scientific Visualization* 10 (4), 13–29 (2018). doi: 10.26583/sv.10.4.02
- [12] A.G. Podvesovskii, R.A. Isaev, *Constructing Optimal Visualization Metaphor of Fuzzy Cognitive Maps on the Basis of Formalized Cognitive Clarity Criteria*. *Scientific Visualization* 11 (4), 115–129 (2019). doi: 10.26583/sv.11.4.10
- [13] R.A. Isaev, A.G. Podvesovskii, *Verification of Cause-and-Effect Relationships in Cognitive Models Using Visualization Metaphors of Fuzzy Cognitive Maps*. *Scientific Visualization* 12 (4), 1–8 (2020). doi: 10.26583/sv.12.4.01
- [14] R.A. Isaev, A.G. Podvesovskii, *Improving the Cognitive Clarity of Graph Models of Knowledge Representation and Decision-Making Using Visualization*. *Ergodesign*. 1 (11), 27–35 (2021). doi: 10.30987/2658-4026-2021-1-27-35
- [15] V. Kasyanov, E. Kasyanova, *Information Visualization on the Base of Graph Models*. *Scientific Visualization* 6 (1), 31–50 (2014).
- [16] V.B. Sylov, *Strategical Decision Making in Fuzzy Environment*. INPRO-RES, Moscow, 1995 (in Russian).
- [17] A.G. Podvesovskii, D.V. Titarev, R.A. Isaev, *Fuzzy cognitive models in software projects analysis and planning*. *Herald of Computer and Information Technologies*, 8, 22-31 (2019). doi: 10.14489/vkit.2019.08.pp.022-031
- [18] A.A. Zakharova, A.G. Podvesovskii, R.A. Isaev, *Mathematical and Software Support for Cognitive Modeling of Semi-structured Organizational and Technical Systems*. in: *CPT2019 International conference Proceedings*, pp. 131–141. Pub. NNGASU and SRCIPT, Nizhniy Novgorod (2019).