Utilizing the Model Graphic Structures when Teaching Generalized Nets

Vassia Atanassova

Institute of Information Technologies, Bulgarian Academy of Sciences
2 Acad. G. Bonchev Str., Sofia-1113, Bulgaria
vassia.atanassova@gmail.com

Abstract

The present paper is devoted to a topic from the newly developed didactic aspect of the modelling of parallel and concurrent processes by generalized nets. Our aim is to explore how the graphic structures, which serve to visualize the generalized net models, can be effectively utilized to enhance the process of teaching / learning of generalized nets. The paper presents two kinds of didactic tasks that have been implemented in already designed training tests, and it analyzes the students' performance and feedback. Three other ideas for test problems are suggested, which place the generalized net model graphic structure in the center of the didactic task. Having in mind that generalized nets is a concept embracing the concepts of Petri nets and all of their own modifications, the ideas presented in this work can be adapted to the didactic ds of other training courses on Petri nets, etc.

Keywords: Bloom's taxonomy, Didactics, Generalized nets, Petri nets, Visualization.

1 Introduction

The present paper is devoted to a topic from the newly developed didactic aspect of the modelling with generalized nets (GNs); i ims to explore how the graphic structures of the GN models can be effectively utilized to enhance the process of teaching GNs [1, 2]. We have already observed in [3] that the theory of GNs is rich in definitions and theorems, providing umerous ways to assess the students' cognitive levels of knowledge and comprehension and predispose to reaching the higher cognitive levels of application, analysis, synthesis and evaluation that are needed for practicing of effective ical modelling.

An entertaining and challenging, yet poorly explored way to do so is to utilize the graphic structures of GNs and place them in the center of some of the didactic tasks.

Based on our survey of the literature on GNs, we can claim that the graphic structures of the GN models do not play the central role in the process of modelling by GNs. The graphic structures play only an role of illustrating the model, as they do not contain (and present) all of the indispensable information about its functioning, which is provided by other accompanying means: textual descriptions of tokens' characteristics, formulas, functions, index matrices. Moreover, none of the information about the graphic structure, which has been coded in the model's XML schema, will later be used when running a simulation of the model. This information is only needed when the user interacts with the GN simulator, using its graphical interface to easily design and edit the model.

However, the graphic structure of the GN model is its st immediate, user- and learner-friendly representation, since it offers a first-glance understanding of the model's consecution and its relative complexity, expressed in the numbers of transitions, numbers of places (some of which being inputs and outputs), number of arcs (some of them looping back), ble bottlenecks (i.e. threats of clogging), as well as possible repetitive patterns (i.e. opportunities of simplification and optimization). So, the graphic structure may not be the "brain", but it is the "heart" of the GN model. Let us discuss on the ways which it can be exploited for the sake of more effective training of GNs and assessing the students' level of comprehension.

2 GN Problems Related to Graphic Structures. Analysis of the Didactic Goals and Errors

In general, there are two kinds of didactic tasks related to the graphic schemes of the GN models:

- case-based didactic tasks, i.e. ones, which are accompanied by a case study with information about a particular modelled process and a problem; and
- independent ones, i.e. didactic tasks assessing the students' general knowledge of GNs by means of their graphic structures.

As we can easily conclude, the independent didactic tasks are much easier than the case-based ones, because they are oriented to the lower cognitive levels of *Knowledge* and *Comprehension*, while the case studies presume these levels being covered and test the higher cognitive levels of *Application*, *Analysis*, *Synthesis* and *Evaluation*, as stipulated by Bloom's taxonomy [4]. In addition, as we will see, the didactic tasks related to GN graphic structures can also train and test the students' watchfulness and wits in a way that other GN training and testing formats are unable to offer.

So far, two kind of problems on GNs have been formulated, aimed at testing students' levels of knowledge and comprehension of the GN graphic structures and the graph operators, applicable over them. These have been approbated in real-life examinations and here the results will be reported for the first time. In the present article we will also offer several other kinds of problems for exercise and testing of GNs.

2.1 Multiple Choice

The first sort of problems was inspired by the so-called multiple choice tests. It is of the independent kind, since the graphic structures are not accompanied by case studies of particular modeled processes. The students were offered four relatively simple GN model graphic structures, among w either exactly one is mistaken, or exactly one is correct. The students were asked to discern the mistaken (correct) GN model and back up their choice with an explanation where the mistake(s) is (are). Correct answers without provided explanation were not taken into consideration. An example of this kind of questions is given on Fig. 1.

Point out the correct GN model graphic structure and explain your answer:

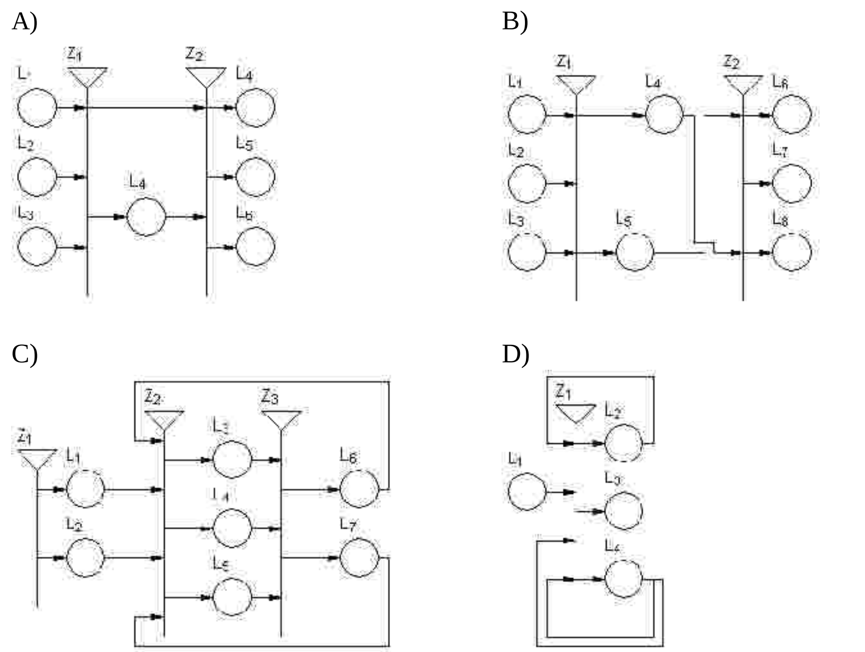


Fig. 1. A sample multiple choice test item with one correct answer and three distractors.

The students are expected to be able to point out the GN model among the three mistaken ones and back up their answer by explaining the mistakes. The correct answer is B) despite of the intersecting arcs connecting places L_4 and L_5 with transition Z_2. There are two errors in A): first, the arc directly connecting transitions Z_1 and Z_2 without passing via a place in between and second, there are two places labeled L_4. The mistake in C) is the lack of input place for transition Z_1, and thus for the model in general, while the mistake in D) is the double arc branching out of place L_4.

The following table presents the distribution of the students from the three trained groups into three groups, with respect to their answers. It is noteworthy, that the numeration of the groups follows the chronological order in which these groups of students were trained and examined, so the feedback was taken into consideration.

	Correct answer		**Partial answer or with mistakes**		**No answer**		**Total**
Group 1	3	*27,27%*	1	*9,09%*	7	*63,64%*	11
Group 2	9	*56,25%*	5	*31,25%*	2	*12,50%*	16
Group 3	11	*100%*	0	*0%*	0	*0%*	11
Total	23	*60,53%*	6	*15,79%*	9	*23,68%*	38

The following classes of common mistakes were registered.

- *Mislabeling of the places and transitions.* Some students do not consider the labels on the places and transitions as *unique* object identifiers. For instance, they may not pay attention to the fact that two different places in the net are labeled in the same way. Others may get mislead by the fact that there are omissions in the serial numbers of the identifiers, fo instance, L_1, L_2, L_4, L_5 with L_3 being omitted has been pointed out as a mistake, which of course it is not. These mistakes indicate that the students are not fully aware how to interpret the GNs in other, better known terms of programming.
- *Missing net components or extra net components.* Some students forget that the basic definition of the GNs states that two transitions in the net may only be connected via a place, and two places may only be connected via a transition. Others fail to recognize when there are two or more arcs branching out of a place or do not figure out which is the “legal” way of drawing a feedback (loop) arc, connecting it with the side of the transition, which denotes the outputs rather than with the left side, denoting the input of the transition. Some students do not interpret well the GN definition which states that each transition must have at least one input and at least one output place and would not see any problem with transition Z_1 on C). This class of mistakes can be easily overcome by clarifying the definition of the GNs and the rules for arranging the net components.

- *Inflexible understanding of the model design.* Some students exhibit surprising inflexibility related to the design of the GN model's graphic structure, and prove unable to think about it "out of box". They tend to attach undue importance to the actual location of the places and arcs, rather than focus on the relations between them hich are graphically expressed. Some of them reason that if an input arc an an output arc are placed on the same level along the transition's length (see the arcs from/to places L_1 and L_4 in example B) above), then the tokens coming from thi input arc are obliged to continue their movement along the above mentioned output arc of the transition. This line of reasoning is of course wrong, but it is indicative as it shows the missing relation between the GN model's graphical structure and the index matrices of predicates for each model's transition which dictate the actual possible routes of token transfer within the net, depending on the logic of the modeled process.
- *Wrongful implications of model's dynamics, temporal and memory components.* As we stated above, the currently discussed sort of GN problems is independent of any case studies of modeled processes, i.e. students are not provided with any further information about the model samples (process logics, tokens characteristics, timings, etc.), except their graphical structures. Hence, it is surprising when some students claimed that the alleged mistakes in the samples were related to presence or absence of tokens within the places, or certain non-true values of the predicates. There were a few students who even speculated about certain transitions and places in the graphic structure, which according to them have been surplus and needed to be removed from the model. This class of mistakes is indicative of the misunderstanding of the difference between the logic of a processed model and its graphic visualization, between the static and the dynamic nature of the GN models, as well as misunderstanding of which information in what form is stored and how can be derived or concluded.

2.2 Application of the Graph Operators over a GN Model Structure

The second kind of problems, included in the GN tests gned so far, is also independent on case studies, but in contrast with the multiple choice task in 2.1., it is a constructive task, hence, a more difficult one and referring to the higher cognitive level of *Application*, according to the Bloom's taxonomy of cognitive goals. It exercises or tests the students' comprehension of the application of the three graph operators, which juxtapose to the net:

- a bipartite directed graph whose sets of elements represent the set of places and the set of transitions;
- a directed graph whose set of elements represents the set of places only; and
- a directed graph whose set of elements represents the set of transitions only.

An example of this test item is given on Fig. 2.1. and the answers are on Fig.2.2.

Apply the graph operators over the following generalized net:

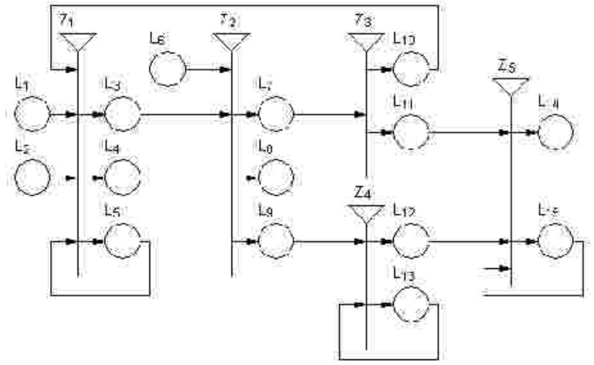
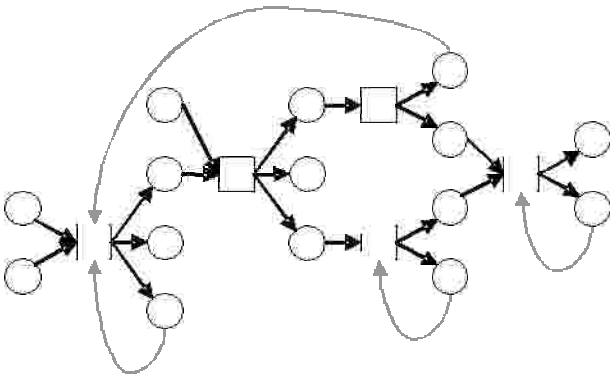
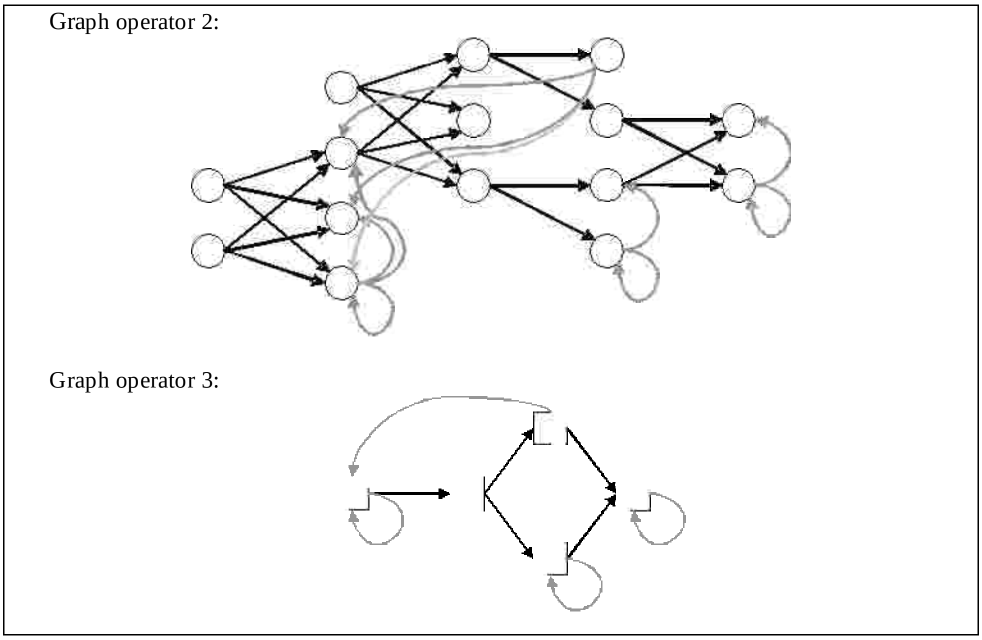


Fig. 2.1. A sample constructive test item for application of procedures for transformation of a GN model into three different graphs, focusing on different sets of elements within the static structure of the net.

In general all problems of this sort exhibited 5 transitions and 14 or 15 places, and 3 or 4 backward arcs. These questions examine the nts' understanding of graphs (the GN graphic structure is a kind of bipartite directed graph) and the ability to correctly trace all the existing relations the places and transitions. The correct answers to the example above are, as follows:

Graph operator 1:





Graph operator 2:

Graph operator 3:

Fig. 2.2. Answers to the constructive test question on Fig 2.1.

Again, the table below presents the distribution of the students from the three trained groups into three groups, as follows.

	Correct answer		**Partial answer or with mistakes**		**No answer**		**Total**
Group 1	1	*9,09%*	2	*18,18%*	8	*72,73%*	11
Group 2	5	*31,25%*	8	*50,00%*	3	*18,75%*	16
Group 3	5	*31,25%*	4	*36,36%*	2	*18,18%*	11
Total	11	*28,95%*	14	*36,84%*	13	*34,21%*	38

As we can see, the success levels on this problem were lower than the success levels on the multiple choice test items. This was an expected effect, and that is why the problems of the second kind scored 3 points in the final results, compared to 1 point for a correct and well substantiated answer to the question of the first kind.

The following classes of common mistakes were registered.

- *Mistakes related to graphs.* Some of the currently tested students in GNs have studied neither graphs, nor finite automata, which naturally resulted in certain typical mistakes in comprehending this task and solving it. It was a general error when applying the first graph operator not to discern between the places and the transitions in the GN and represent them as (visually) different sets of vertices in the bipartite graph, or mix the specificities of the first and the second operator. An even more common mistake was the lack of the edges' orientation, despite they seemingly knew that all the three resulting graphs were directed ones.

- *Mistakes in building the link between GNs and graphs.* Some students showed less ability in relating the concepts of GNs and graphs when representing one and the same abstract model in the terms of these two different areas. For instance, many students missed drawing the backward (looping) arcs in all three graph, especially the second one (this was the most common mistake). Others simply ignored parts of the model, which they found repetitive, or, vice versa, added some extra vertices and edges to the graphs, which were not present in the original GN model.

2.3 Redrawing a Model

A simple constructive exercise is related to redrawing a sample GN model in a way that a predefined condition is accomplished.

For instance, let us consider the initial GN model (a) on Fig. 3. The students are required to redraw it in a way that transition Z_4 is located in the leftmost part of the new graphic structure (where it is most natural to be seen, having L_6 as the only input place for the net, see Fig. 3 (b)). The same exercise is required for all of the rest transitions in the net, respectively shown on Fig. 3 (c) and (d). This exercise, as easy as it may seem, is helpful in provoking students to search for the semantics behind the model, to discover the functional relations between its components, to flexibly rearrange the model's graphic structure in semantically synonymous ways, and not obey unnecessary design conve As we can easily check, by redrawing the model with any other transition to the left, we do not intervene in the logic or consequence of the modelled process, in the index matrices of predicates or capacities, in the priorities, timings, characteristics, etc components of the model.

2.4 Transforming Petri Nets and their Modifications into Generalized Nets

A new class of interesting problems in the training of GNs and GN modelling is related to revealing the links between GNs and the Petri nets. This is especially useful when training students who have background knowledge of Petri nets, as well as when presenting the GN theory in front of established experts in Petri net modelling. However, these problems are inapplicable in cases when the trainees have no preliminary knowledge of Petri nets, which is often the case when they are specialists in some side field of practical GN application, for instance medicine, chemistry, administration, and many others, [5]. A particular example will be discussed in the following section.

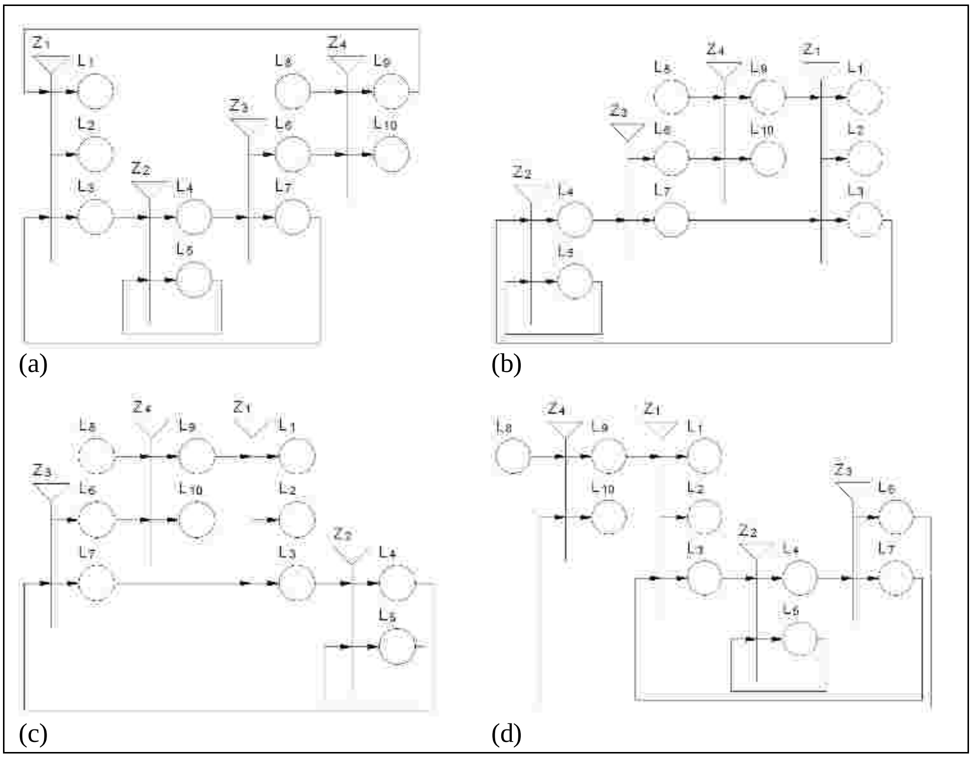

(a)
Z1 Z2 Z3 Z4
L1 L2 L3 L4 L5 L6 L7 L8 L9 L10
(b)
Z1 Z2 Z3 Z4
L1 L2 L3 L4 L5 L6 L7 L8 L9 L10
(c)
Z1 Z2 Z3 Z4
L1 L2 L3 L4 L5 L6 L7 L8 L9 L10
(d)
Z1 Z2 Z3 Z4
L1 L2 L3 L4 L5 L6 L7 L8 L9 L10

Fig. 3. A sample constructive test item for redrawing a GN model.

2.5 Case Studies

Studying cases and solving certain simple problems has not yet been implemented in the training courses on GNs, but it is undoubtedly a very challenging and practice oriented task. Given the verbal description of a case, the students can be asked to draw the graphic of the corresponding GN model, write down the index matrices that correspond to each the transitions in the model, explain the initial tokens characteristics and functions that assign new characteristics in the course of the model functioning, and express any consideration they may have about the temporal conditions for activation of the transitions, or priorities of the transitions, places and tokens, or capacities of the arcs and places, etc. In case that the students have preliminary knowledge of Petri nets or some of their modifications, they may even be hinted with the graphic of the respective modeled problem, as drawn in these terms.

Different solutions may be expected here, yet with equivalent modelling abilities, and the difference may be not merely on design level, as discussed in the previous section, but it may be on a higher level, expressed in different sets of transitions and places, different index matrices of predicates, different sets of tokens' characteristics and functions. In fact, the question of "models synonymy", as discussed in the context of generalized is a challenge that has not been explored as of today. It will be interesting to compare the resulting GN models and make certain assumptions and/or conclusions about how do structural differences affect the model performance, if the logic of the modelled process remains intact. For instance, such effects can be sought in directions like memory and resource allocation, time for calculation, etc. A more detailed survey on this case-based didactic approach with the respective results from training and examination, as well as discussions on the newly raised questions, will be the subject of a future author's research.

3 Conclusions

The present paper is devoted to one of the aspects of training generalized nets, namely employing the visual information from the GN models' graphical structures and linking it to the other sources of info in the model (index matrices, token characteristics, functions and others). This visual approach with its opportunity for immediate representation and comprehension of the GN theory is especially helpful in trainees on beginner level or ones who have no preliminary background in artificial intelligence, mathematical modelling or programming but need this apparatus to solve practical problems from the their field of competence.

The first two of the presented classes of problems were recently developed and have been already approbated with 38 students from three Bulgarian universities. The problems were part of a newly designed approach to formative and summative testing which will be further elaborated from now on, targeting various groups of learners with their different educational and practical needs. It has been considered very important to analyze the didactic goals meant with each of these classes of problems, as well as the cognitive levels reached, per Bloom's taxonomy. For the first two kinds of problems this analysis and justification is done only now, after they have been approbated in real-life situations, but for the rest three ideas presented this analysis is herewith done in advance, as required by the traditional design methodologies for standardized tests. However, an interesting point of this communication shall be considered the presented error analysis which offers a useful feedback that shall be taken into consideration for the future lecture courses on generalized nets.

The third newly presented idea for training the students in generalized nets by case studies opens a brand new direction of research within the methodological aspect of the theory of generalized nets. Thus, practical evidence is given that the newly proposed didactic aspect of the theory outreaches the classrooms, adding value to the theoretical research in the field of the generalized nets.

Acknowledgments

This work is partially supported by the National Science Fund of the Ministry of Education, Youth and Science of Bulgaria under Grant DID-02-29 and by the Ministry of Science and Higher Education of Poland, Grant Nr N N519 384936.

4 References

[1] Atanassov, K., *Generalized Nets*. World Scientific, Singapore, 1991.

[2] Atanassov, K., *On the Theory of Generalized Nets*. “Prof. M. Drinov” Acad. Publ. House, Sofia, 2007.

[3] Atanassova, V., Design of Training Tests on Generalized Nets, *Proc. of 5th International IEEE Symposium ‘Intelligent Systems’*, 7–9 July 2010, London, UK, 327–330, 2010.

[4] Bloom, B. S., *Taxonomy of Educational Objectives*. Allyn and Bacon, Boston, MA, 1984.

[5] Alexieva, J., Choy, E., Koycheva, E., Chapter 8 Review and bibliography of generalized nets theory and applications in *A Survey of Generalized Nets*. Choy, E., Krawczak, M., Shannon, A., Szmidt, E. (eds.) Raffles KvB Monograph, No. 10, Sydney, 2007.