USING GENERALIZED PETRI NETS FOR E-LEARNING SYSTEM MODELLING*

D. Vynikarová

Czech University of Life Sciences, Faculty of Economics and Management, Department of Information Engineering, Prague, Czech Republic

The aim of this paper is the design of a generalized petri net and object connection modelling rules. The suggested generalisation is appropriate for intelligent system modelling, especially for e-learning systems, which monitor learning and evaluation process of students.

intelligent system; Petri net; generalized Petri net; semantic information theory; e-learning

INTRODUCTION

This paper focuses on intelligent systems as systems supporting information activities The term information is interpreted as a semantic information, in the following way: The individual crisp finding about the point x of the given universe U is characterised by the (crisp) subset $\delta(x) \subseteq U$ of the universe. The information about the point x of the universe is the set $J(x)$ of subsets of the universe U such that the following is true:

1. $\delta(x) \in J(x) \Rightarrow \delta(x) \neq \emptyset$,
2. $\delta(x) \subseteq \delta'(x)$ and $\delta(x) \in J(x) \Rightarrow \delta(x) \in J(x)$,
3. $\delta(x) \in J(x)$ and $\delta'(x) \in J(x) \Rightarrow \delta(x) \cap \delta'(x) \in J(x)$.

In other words, information on the member of universe is the filter on the universe (see Vaníček et al., 2007). The fuzzy information can be described by the filter of fuzzy findings, the functions defined on the universe and mapping the universe into the closed interval <0, 1>. The fact that the value of the membership function, which represents the fuzzy finding $\delta(x)$, is equal to p is dented by $(p)\delta(x)$. Therefore $(1)\delta(x)$ denotes $x \in \delta$, $(0)\delta(x)$ denotes $x \in U \div \delta$.

Information activities realized by the intelligent system can consist of collecting, processing and amiability interposing of information by means of data storing, data transformation and data communication. Intelligent system is composed of sensors and executive elements for the contact with the subject area, which is processed by the system. Intelligent systems have made the difference between object on the subject area and relations between objects and have to support a dynamic of the system evaluation.

To describe the intelligent system dynamic Petri nets are often use as a mathematical model. The limitation of Petri nets using for this purpose is the fact, that Petri nets cannot make a difference between the request propagation from the customer into the proper part of the system, where the required information is stored and the process of the request disposal from the source of information to the customer. The same situation is typical in e-learning when the student is navigated to the source of information, which is required for solving the individual problem. For this purpose some generalization of Petri net seems to be useful. The Formal description of such a generalisation of Petri net concept is the intended merit of this paper.

Proposed generalisation of Petri nets can be used for intelligent system modelling in general, especially for e-learning systems, which monitor learning and evaluation process of student.

MATERIAL AND METHODS

Intelligent system

The term intelligent system will be used without is not a reference to a human, group of people, machine or a system of people and machines. Typically intelligent systems can be composed from the following elements (basic intelligent systems).

- **Intelligent sensor –** this system captures finding of type $(p)\delta(x)$ for single objects (subject or phenomenon) from the real world. The x is element (point of the universe) and δ is some characteristic of elements from the real world, δ is subset of the universe U. Number $p \in \langle 0,1 \rangle$ is a level of certainty, that element x has a characteristic δ. Finding $(0)\delta(x)$ (it is written as $\delta(x)$ or $\delta'(x)$ sometimes) means certainty, that x has not characteristic δ ($\delta \notin x$ or $x \in U \div \delta$). Intelligent sensor can be realised by human, group of people, machine or people by the help of techniques. Intelligent sensor is able to join measured value with an object, which is related to its data. It is a principle of intelligence. The thermometer itself is not an intelligent sensor. But if

* The paper is supported by the grant project No. 2C06004 – Information and knowledge management – IZMAN and 1M06014 MŠMT ČR.

the thermometer measures the time and its place and measured value is joined with the semantic indicator of place and time, thermometer is in role of intelligent sensor.

- **Intelligent element** – this system realizes finding of the object on the basis of finding $(p)\delta(x)$, which are about this object known. It can be a human, group of people, machine or system of people and machines – for example every production, book finding by the help of librarian or cashier operations.
- **Data store** – the intelligent system activity consists in the file management, database and document management, which represent finding about objects and accessing these findings to users. Data in the form $(p)\delta(x)$ is associated with an object, which is related to object in a data store. The data store output is controlled by external requirements.
- **Finding converter** – the converter is a system, which creates the finding about objects from set Y on the basis of a finding about the set X. The converter can be a human who takes decisions on the basis of finding. The converter can be a specialist or a team, which solves some problem or it is a designer or engineer's office, the car driver, the pilot or technical equipment, which regulates the process plan or every computer program, which executes an algorithm. A converter transfers a finding of type $(p)\delta(x)$ for $x \in X$ to findings type $(q)\beta(y)$ for $y \in Y$. $X = Y$ is a special but frequent case. A converter is able to call a compiler in this case.

It is possible to create more complicated systems from the previous four intelligent systems. For example:

- **An expert system** – the expert system is a net of the intelligent systems, which is composed of the sensors, data stores and converters. This net can be multi-level and hierarchical. There are no executive elements on the lowest level.
- **A management system** – the management system is a net of intelligent systems, which always contains the sensors and the executive elements.

The subject area must be defined for every intelligent system. This subject area is not a part of the system, but system action is bind with this subject area. This area is created from a given objects and relations between them. The universe is created from this area and the findings and the information are defined there.

The Petri nets

In the intelligent system that works in some subject area, a difference between the objects and object relations has to be made. Therefore it is possible to use a diagram with two types of nodes and the relations between them. One type of nodes represents objects; the other type of nodes represents relations between them.

The changes are happening in the time or in the information area. They are represented by moving the marks through the diagram nodes. This principle is described via standard Petri nets.

The Petri net is defined as bipartite and directed multi graph $G = (P, T, F)$, where:

- $P = \{p_1, \dots, p_m\}$ is a nonempty set of nodes called places (positions),
- $T = \{t_1, \dots, t_n\}$ is a nonempty finite set of transitions,
- F is a set of oriented edges and flows,
- $P \cap T = \varnothing$ for each item of the set F is defined an ordered pair of nodes where the one of them is from P and the second one is from T (see Češka, 1994).

The marking of the net $G = (P, T, F)$ is a function μ: $P \rightarrow N$, where N is a set of non-negative-integers. The marking $\mu(p_j)$ is represented by the relevant mark count (see Girault, Valk, 2003).

The evolution of the marked net (P, T, F, μ) is a process of the marking change the stage using the following rules: If all input positions of a transition $t \in T$ are nonzero marked, the transition will be started. The result of this transition is a marking change only in input and output positions of the transition i.e. the marking are decreased by 1 in all input positions and are increased by 1 in all output positions.

- If two or more transitions can be started and these transitions have not any common input position the transitions will be executed in parallel or in arbitrary sequence.
- If more than one transition can be started and the two of them have any common input position then only one of these transitions will be executed. This situation is called conflict. If any external rule does not resolve the conflict, it will lead to non-determinism.

Let $X_1, X_2, \dots X_m$ are sets of objects. Then any subset of Cartesian product $X_1 \times X_2 \times \dots \times X_m$ is a relation with domains $X_1, X_2, \dots, X_m$.

The modelling rule 1:

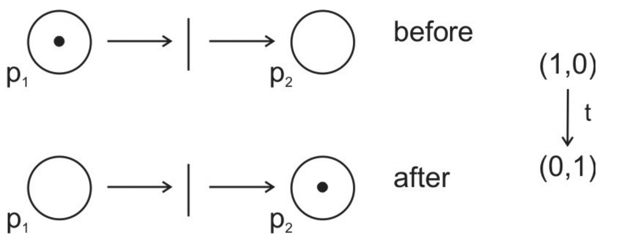
For each object x from a subject area is defined a position in the Petri net with only one mark. The position without any mark means a set X of objects, which are linked to this position. One mark in the concrete position means a concrete object $x \in X$.

The modelling rule 2:

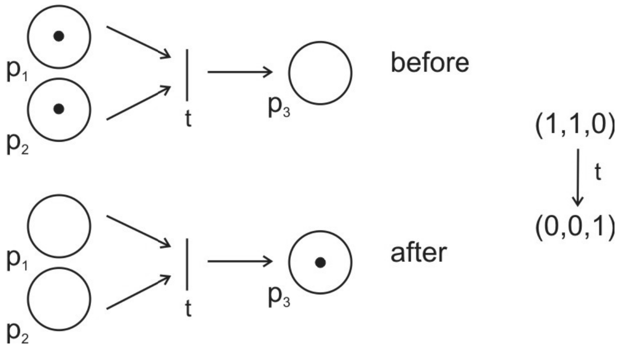
Every object relation in a subject area represents one or more relations and incident edges, used in these relations in a Petri net. Objects $x_1 \in X_1, x_2 \in X_2, \dots, x_m \in X_m$ are represented by the relation between the places, i.e. a matrix $(x_1, x_2, \dots x_m)$, which is an element of a relation R with domains $X_1, X_2, \dots, X_m$.

It can be represented by a functional dependence:

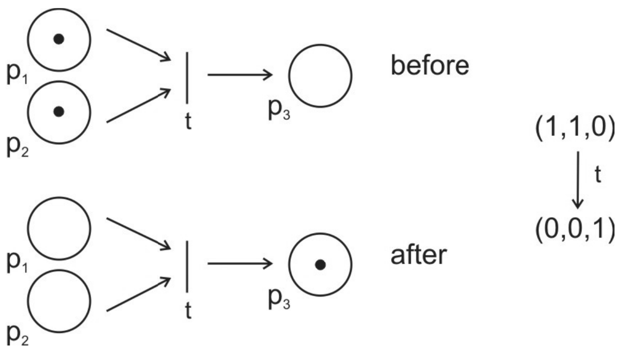
- One variable function:



- Two variable function:



- Function with alternative and non-deterministic character:



RESULTS

Generalized Petri net

It is useful to generalized Petri net to be able to take more process with interrelations. For example:

- The producer (of information) – the consumer
- The study (learning) – examination

The generalized Petri net is defined as bipartite and directed multi graph $G = (Q, S, H)$, where:

- $Q = \{q_1, \dots, q_m\}$ is a nonempty set of nodes called places (positions)
- $S = \{s_1, \dots, s_n\}$ is a nonempty finite set of transitions
- $Q \cap S = \emptyset$
- H is a set of oriented edges. Every edge is assigned the ordered pair of nodes, exactly one edge is Q and exactly one is S. It can be represented by two incidence matrix (m, n) and (n, m).

The placing of the first type in the generalized Petri net $G = (Q, S, H)$ is a function $\chi : Q \cup S \rightarrow D = \{0,1\}$. The fact, that $\chi(v) = 1$ is marked by a symbol *.

The placing of the first type is defined by a $m + n$ element binary vector $(\chi 1, \dots, \chi_m, \chi 1, \dots, \chi^n)$. **The function of the first type** for a placed generalized Petri net means a change of the first type marking according to these rules:

- If a transition has zero marking of the first type (there is no *) and at least one of its output position is marked by * the transition of the first type will be done. The transition gains a value $\chi(s_j) = 1$ and all its input positions gain (or stay on) a mark *. The marks of output positions of the transition and marks of all other nodes stay without change.
- If more than one transition fulfils the first rule then all transitions will be executed in arbitrary sequence.
- The transitions are executed until at least one fulfils the first rule. Then a subnet $Q^* \subset Q$, $S^* \subset S$, $H^* \subset H$ of nodes (positions and transitions) with χ values 1 (marked by *) is defined.

The placing of the second type in the generalized Petri net $G = (Q, S, H)$ is a function μ: $Q^* \rightarrow N$ (N is a set of non-negative integral numbers) which transforms nodes marked by * (i.e. only the subnet defined by the placing of the first type) to the set N. The values of $\mu(v)$ are graphically marked by the relevant count of dots (•). The dots can be only with stars (*) so the star is not written – the dot is stronger than the star.

The function of the second type goes on the subnet marked by *. It starts in input marking according changed rules of the Petri net – When • is filled in the * will be erased. When • is removed the * will not be returned. The dots (•) are only filled in positions marked by *.

The modelling rule in the generalized Petri net $G = (Q, S, H)$ tells that the placing of the first type and the second type go concurrently but the placing of the second type is preferred. The marking by • ($\mu(v) > 0$) removes the marking by * ($\mu(v) > 0 \Rightarrow (\chi(v) = 0)$.

- When a transition of the second type is processed the transition mark * will be changed.
- When a transition of the second type is processed the marks * will be erased in all output positions of the transition. If the transition have not any output position marked by * the mark of the transition is erased too.

Generalized Petri net using for an intelligent (e-learning) system modelling

Rules for an intelligent (e-learning) system modelling using the generalized Petri net:

- For each information (data) source is defined a position in the generalized Petri net.
- For each system which transforms data of objects from one set to data of objects from another set is defined a transition. The information flows in the direction of oriented edges (→).

- The marks * are used for representation of requirements. The marks •″are used for representation of answers (accomplishment of requirements).
- The requirements (*) are assigned to positions (data sources) and to transitions (transform systems). Marks • are assigned only to positions.

The intelligent system analyzes requirements and tries to complete them. This process can be described by the change of the generalized Petri net states:

- Questions or requirements are represented by *. Answers are represented by •.
- The mark *, which represents some attribute of an object from universe, is inserted into the generalized Petri net from some data store or from some executive element or from outside. The answer is created in the position marked by *.
- If sufficient information (information which accomplishes the requirement) is in this position the mark * will be changed to mark •. The mark • is defined by type of an answer $(p)\delta(x)$ which is the best expressed by data $\Delta(\mathrm{x}) = \{(p_1)\delta_1, (p_2)\delta_2, \ldots, \}(x)$.
- If information is not sufficient in the position the mark * will stay and will be filled in all input transitions of this positions in opposite of the direction of edges.
- The requirements can be transformed parallel or gradually. If sufficient information is in the position the mark * will be changed to mark •. If not the process will continues recursively until the mark is change to • at least one position.

These rules can be used to manage a student to (repeatedly) study and practice a part of study document to be able to acquire further knowledge and required habits off it.

Description of example in the pictures:

A requirement enters into the data store q_7 (Fig. 1). In the data store there is not sufficient information for answer.

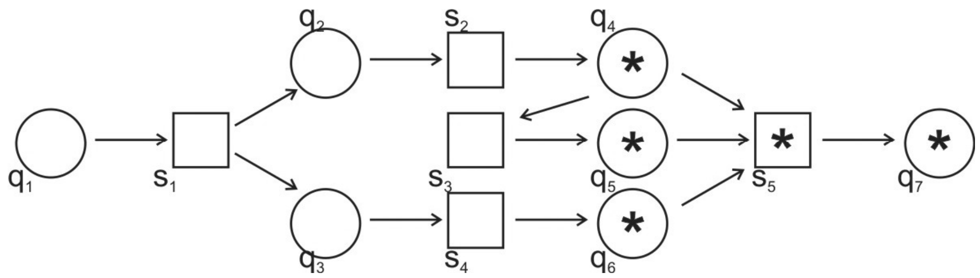


Fig. 1

The requirement is expanded to a transformer s_5 (Fig. 2). The transformer asks q_4, q_5, q_6 for needed information (verifies student's knowledge of preconditions).

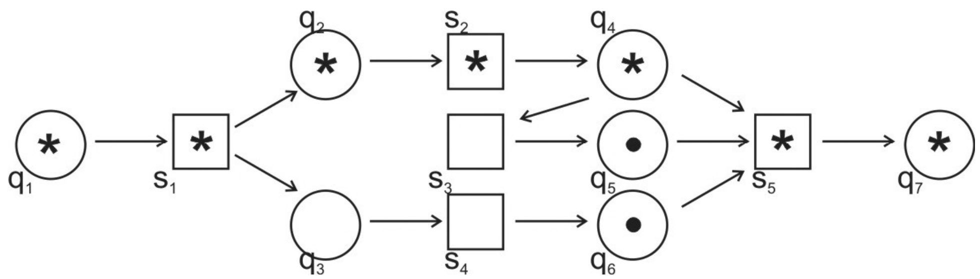


Fig. 2

If the sufficient information is in q_5 and q_6 (for s_5) the mark * will be changed to mark •. If sufficient information is still missing in q_4 the mark * is expanded from q_5 to the transformer s_2 and its input data store s_1. If sufficient information is still missing in q_2 the mark * is expanded also to the transformer s_1 and data store q_1 (Figs 3 and 4).

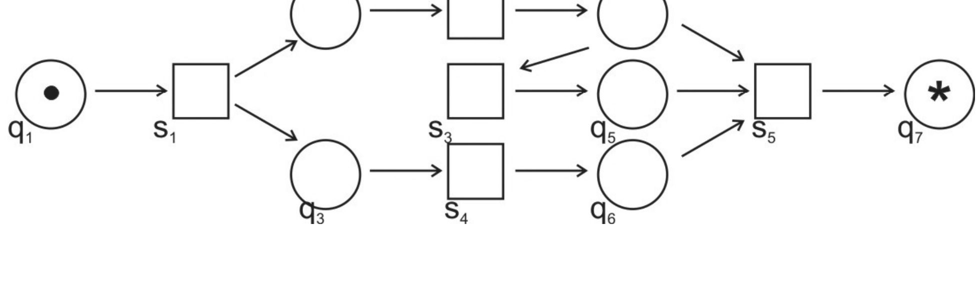
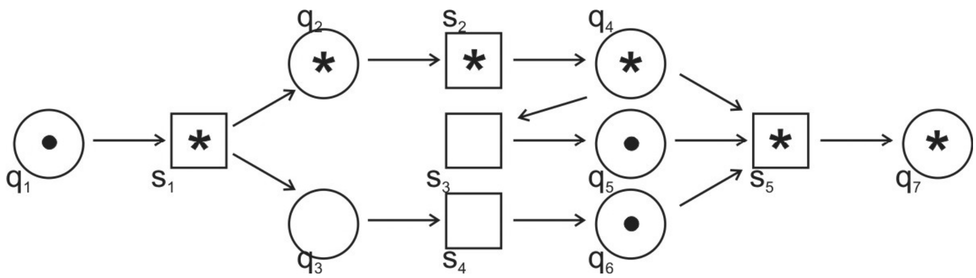


Fig. 3

Fig. 4



If the sufficient information is in q_1 the reverse process will be started. The marks * are changed to marks • successively only in the subnet of marked positions (Fig. 5). The process is finished after the transfer of mark * to the position q_7 of the original requirement (Fig. 6).

Fig. 5

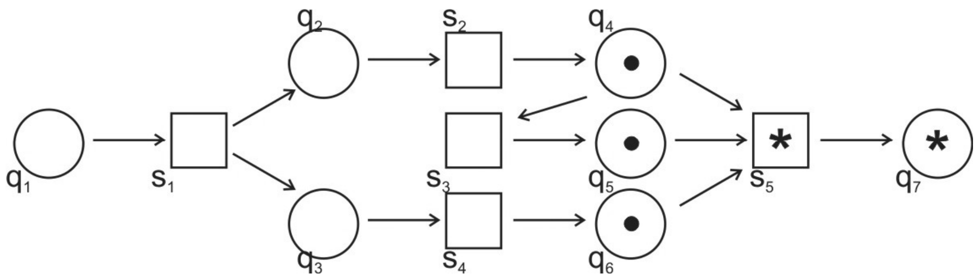
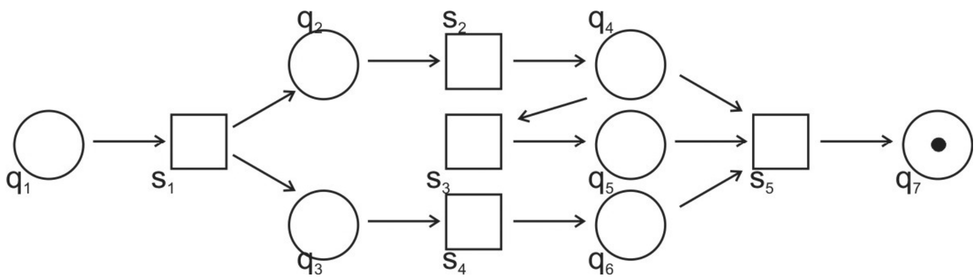


Fig. 6



CONCLUSION

It is possible to use the generalized Petri net for description of several processes with interrelation. In the beginning the subnet is defined which is executed according to generalized modelling rules of the Petri net. The generalized Petri net and the rules can be used for modelling of intelligent systems especially of information transformers. The intelligent system analyzes requirements and tries to complete them. This process can be described by the change of the generalized Petri net states.

REFERENCES

ČEŠKA, M.: Petriho sítě. Brno, CERM Publishing 1994. 94 pp.

GIRAULT, C. – VALK, R.: Petri nets for systems engineering. Berlin, Springer Verlag 2003.

VANÍČEK, J. et al.: Teoretické základy inoformatiky. Praha, Kernberg Publishing 2007.

Received for publication on January 14, 2008
Accepted for publication on March 3, 2008

VYNIKAROVÁ, D. (Česká zemědělská univerzita, Fakulta provozně ekonomická, katedra informačního inženýrství, Praha, Česká republika):

Využití zobecněných Petriho sítí pro modelování e-learningových systémů.

Scientia Agric. Bohem., 39, 2008: 102–107.

Informace o předmětné oblasti tvoří součást inteligentního systému, který s ní pracuje. Tyto informace se v rámci informačního systémů shromažďují, zpracovávají a zpřístupňují pomocí datových skladů a transformátorů. Inteligentní systém obsahuje čidla a výkonné prvky (realizátory) pro styk s předmětnou oblastí. Inteligentní systémy pracující v nějaké předmětné oblasti musí rozlišovat objekty a vztahy mezi nimi. Mezi objekty probíhají změny v čase nebo

v informačním obsahu. Tento princip popisují klasické Petriho sítě. Petriho sítě je vhodné zobecnit tak, aby byly schopny zachytit více procesů, které mají vzájemnou vazbu.

Zobecněná Petriho síť je biparitní orientovaný multigraf G = (Q, S, H), kde:

- $Q = \{q_1,\dots,q_m\}$ je neprázdná konečná množina vrcholů zvaných místa
- $S = \{s_1,\dots,s_n\}$ je neprázdná konečná množina přechodů
- $Q \cap S = \varnothing$
- H je množina orientovaných hran takových, že každé hraně je přiřazena uspořádaná dvojice vrcholů, z nichž právě jeden je z Q a jeden je ze S.

Inteligentní systém pracuje tak, že nejprve analyzuje požadavky a pak se je snaží zodpovědět. To lze popsat změnou stavů zobecněné Petriho sítě, kde dotazy (požadavky) se znázorňují značkami * a odpovědi značkami •. Jsou-li v příslušném místě potřebné informace, změní se * na •. Pokud ne, pokračuje se rekurzivně, dokud se na některém místě neobjeví •. Totéž lze využít pro řízení toho, kterou část látky má student znovu studovat a procvičovat pro to, aby měl předpoklady zvládnout další poznatky či získat potřebné návyky.

inteligentní systém; Petriho síť; zobecněná Petriho síť; teorie sémantické informace; e-learning

Contact Address:

Ing. Dana Vynikarová, Česká zemědělská univerzita v Praze, Fakulta provozně ekonomická, katedra informačního inženýrství, Kamýcká 1076, 165 21 Praha 6-Suchdol, Česká republika, tel.: +420 224 383 806, e-mail: vynikarova@pef.czu.cz