A new method for converting extended version of petri nets to fuzzy extended markup language

Solmaz Soltani[a*] and Ali Harounabadi[b]

[a]*Department of computer, Science and Research Branch, Islamic Azad University, Bushehr, Iran*
[b]*Assistant Professor, Department of Computer Science, Central Tehran branch, Islamic Azad University, Tehran, Iran*

C H R O N I C L E	A B S T R A C T
	A model is a simple and straightforward performance of structure and behavior of the system under study expressed as mathematical relations. In fact, with a model, knowledge and information on modeled phenomenon are prepared and the cost and risk of applying actual phenomenon are eliminated. Petri nets, on the other hand, are used to determine the important information on structure and behavior in modeled system and formal models are mainly state dependent requirements. These requirements are significantly indefinite and can be expressed by language variables. If XML labels can express the formal models (Fuzzy Petri net) as the most valid data format, then we will have relationships with other data format. In this study, characteristics of a Fuzzy Colored Petri Net are expressed by fuzzy developed Markup Language labels. Therefore, in addition to fuzzy XML labels, the concept "color" will be used for them. Application and advantages of suggested method will be determined by a case study in bank facility system. The results of the evaluation indicate the advantages of the suggested grammar for supporting elements available in fuzzy formal models.

1. Introduction

One of the ways for investigating a system is to model it using some modeling instrument and the Petri nets are considered as the best modeling instruments for different systems including interrupted and continuous. Petri net is a mathematical description, which presents a graphic or visual performance of system and it can be applied for determining important information on modeled system structure. Conventional Petri nets have been used to model systems with deterministic data but they are often unsuitable to handle vague and uncertain data in practice. In many real world problems, we encounter various systems where there are some degrees of uncertainty and they cannot be modeled using mathematical model. As the most valid data format, XML has the capability of supporting formal models as Petri net. PNML, as a markup language has expressed Petri net in previous researches. Since we observe uncertainty in nature and need to support this subject in

systems, we will apply phase approach in stating the proposed model of this paper. As a developed version of Petri net, Fuzzy petri net has phase concepts. In this net, various elements including places and transitions can be expressed in terms of fuzzy. In this study, Markup Language is presented as a support for this version. Having labels supporting phase concepts, FXML helps us handle this fuzzy formal model. In addition to covering fuzzy concepts in Petri net, it expresses the structure and grammar in Markup Language in the present study. In other words, it is expected to support elements available in fuzzy Petri net in FXML. The purpose of this study is to present a Fuzzy Markup Language in order to support fuzzy colored Petri model. Therefore, generally, the purpose of the present study can be the following:

1. It defines labels in XML, which states the fuzzy concepts.
2. It presents a grammar based on XML labels so that they could support petri net concepts.

We must use fuzzy systems to handle complex systems in which there are some degrees of uncertainty in describing the nature of the system. On the other hand, formal models allow describing dependent needs. XML is selected in order to enter a valid data format and fuzzy petri net is conversed into FXML.

2. Background of the study

In the following, some necessary concepts are described:

2.1 Colored Petri nets

In general, Petri nets are considered as an instrument for studying and modeling systems and there has been a great interest on petri nets for modeling interrupted systems for the past two decades. Petri nets are constituted of two main components: a set of places and a complex of transitions. Formally, a Petri net C has been defined; foursome $C= (P, T, I, O)$ where P is the set of places, T, set of transitions, I, input function and O, output function. Colored Petri nets is believed to be introduced by Kurt Jensen as a developed model of Petri nets. In addition to places, transitions and tokens, concepts "color, guard and expression" are introduced in these nets. Data types in these nets are carried by tokens. Colored Petri nets present precise models of complex asynchronous processing systems. In spite of Petri nets, token are distinguishable from each other in these nets because each token have features called color. In a Colored Petri net, a place can include zero, one or more tokens of a set of colors, which determines type of tokens, where the place can include. Marking a place is as a MultiSet. Transition in colored Petri nets is similar to transition in a Petri net. In colored Petri nets, each token include a series of colors and can take only values defined by the same colors. Arcs are responsible to connect places to transitions (input arcs) and transitions to places (output arcs) and a transition is attributed to each arc. An expression of arc has a form of a MultiSet that determines the following cases:

1. In input arcs, there is a combination of tokens, which must be presented in input place in order to activate transition.
2. In output arcs, there is a combination of tokens where transition is activated in output places.

2.2 Extensible Markup Language

XML is an abbreviation of Extensible Markup Language and Extensible Markup Language is termed a language, which helps user and or other programs to enter necessary information, to edit and or to use required information based on a series of signals and signs. XML is a technology based on character molding and helpful for human. XML does not contain a certain work and has been designed for classification, storage and transition of information.

2.3 Fuzzy Petri net

Petri net has compatibility with classic logic whereas, in practice, we concern with complex systems where there is a degree of uncertainty in description of input data. Therefore, if we wish to model

these systems with Petri net, we have to demonstrate these uncertainties stated based on inexact and vague expressions in Petri model and this requires introducing fuzzy concept in Petri model. In fact, all various methods combining Petri net and fuzzy sets are called Fuzzy Petri net.

3. Related works

According to Kindler (2006), PNML is actually the transformed format of XML for describing Petri net. PNML uses UML supermodels in order to perform and clarify the definition of XML format. According to Brosso et al. (2002) PNML is a type of high-level Petri nets used to simulate systems based on web, electronic activities such as e-business; business-to-business (B2B) and e-commerce and mobile wireless applications in ad-hoc. Combination of Petri net and XML files as an important access can be used to state the characteristics of complex processing commercial duties in distributed homogeneous systems. Gehlot and Nigro (2010) presented vocabularies and structures of colored Petri nets and explained that colored Petri nets could combine the strengths of conventional Petri nets with a high level programming language and it could help us reach suitable modeling of large-scale systems. CPN model is an applicable performance of system analyzed through simulation. In fact, CPN model has been made by CPNTools that is a software graphic instrument and interface used to establish, edit, simulate and analyze models.

Jensen et al. (2007) presented an investigation where colored Petri nets represent validation and modeling language for systems in that concurrency communications and coordination play a significant role. Colored Petri nets are modeling language of an interrupted event and a standard modeling language, which provide primary definitions of types of data, data manipulation and data description. Ma et al. (2010) presented a new fuzzy model is given for XML data according to XML schema. Using this model, fuzzy information in XML documents were shown, naturally and related algebra was formally introduced along this model. XML was a standard method to perform and to exchange information throughout the web and it is considered as a next generation of web language on internet. Gaurav and Alhajj (2006) presented a method to combine fuzzy and inexact data in XML documents. Moreover, by the use of simulation relations and probability theory, some methods were described for introducing ambiguity. Then, an approach was presented in which data saved in fuzzy relational database were appeared as fuzzy XML documents on web.

4. Suggested grammar

The most important tags that are common in describing types of Petri net, including FPN, CPN, PN and their characteristics have been given in the following. The primary difference among various types of Petri net and fuzzy Petri net is in description of their fuzzy behavior and it is also associated with elements, which are fuzzy in nature. Fuzzy elements is expressed by some features added to its characteristics but we need new labels for describing fuzzy behavior that can describe this behavior. These labels and their characteristics are given in Tables 1, 2, 3. Another discussion is that nets can include sub-elements. Different kinds of Pet net indicate various concepts of sub-elements. In fact, sub-elements are secondary elements that exist in Petri nets.

Table 1
Main parts of Petri nets

Name	Characteristics
<net>	id, type, graphics, name, description
<place>	id, graphics, name, description, initial marking, colset
<transition>	id, graphics, description, transformation, guard
<arc>	id, source, target, graphics, name, description, inscription, arc expression

Table 2

Phase parts of Petri nets

Name	Characteristics
<net>	$D=\{d_1, d_2, ..., d_n\}$
<place>	βα
<transition>	CF(μ)
IF	then
THEN	New initialmarking.place ID, Newα.place ID

Table 3

Secondary elements in Petri nets

Name	Characteristics
Page	id, description, THEN, New α, graphics, name

4.1 Main and fuzzy elements of Petri net

Characteristics of the main and fuzzy elements of Petri net are given in the following section.

4.1.1 Petri net ‹net›

A Petri net is defined by the element ‹net›. This element has the following features:

a. id: a unique identity that provides the possibility of reference to net by another net.
b. Type: it is the type of Petri net. In fact, it is the rules defined in Petri net that is defined as a DTD file.
c. Graphics: this is an optional element. It determines place and dimension of general sections of page.
d. Name: it states the name and is optional. In fact, it does not have certain characteristics and are placed in element ‹text›.
e. Description: it is an optional element and contains description of an element.
f. D= $\{d_1, d_2, d_3,..., d_n\}$ is an unlimited set of characteristics.

4.1.2 Place ‹place›

A place is defined with the element ‹place› and has the following features:

a. id: a unique identity that provides the possibility of reference to net by another net.
b. type: it is the type of Petri net. In fact, it is the rules defined in Petri net that is defined as a DTD file.
c. graphics: this is an optional element. It determines place and dimension of general sections of page.
d. name: it states the name and is optional. In fact, it does not have certain characteristics and are placed in element ‹text›.
e. description: it is an optional element and contains description of an element.
f. Initial marking: this element shows multiple set of token colors.
g. $\alpha: P \rightarrow [0.1]$: is a cumulative function. In fact, it is a written form of places according to real value between 0 and 1.
h. $\beta: P \rightarrow D$: is a cumulative function. In fact, it is an objective written form between places and characteristics.
i. colset : it is the element of colors and determines the type of tokens put in places.

4.1.3 Transition ‹Transition›

A transition is defined with the element ‹Transition›. This element has the following characteristics:

a. id: a unique identity that provides the possibility of reference to net by another net.
b. graphics: this is an optional element. It determines place and dimension of general sections of page.
c. description: it is an optional element and contains description of an element.
d. transformation: this element determines the degree of token transformed by a transition. In fact, this element is optional.
e. *CF(μ)*: this element is confidence factor and the more its degree, the more the degree of confidence will be.
f. name: is for identifying the transition.
g. guard: this is a guard element. In fact, it is bully expression, which includes some variables.

4.1.4   Arc ‹arc›

An arc is defined by the element ‹arc› and has the following characteristics:
a. id: a unique identity that provides the possibility of reference to net by another net.
b. source: $\{X_1, X_2, ..., X_n\}$
c. target: $\{Y_1, Y_2, ..., Y_n\}$
d. graphics: this element determines middle points of target layers. For the element ‹arc›, 0, 1 and or more times can be occurred.
e. name: this element indicates name and is optional. In fact, it has no specific characteristic.
f. description: this element is optional and contains description of an element.
g. inscription: this element is optional.
h. arc expression: this element states the arc characteristic. Arc expression includes some variables. In fact, when arc expression was evaluated, its result is a multiple set of token colors.

<IF>

The characteristic of this element is <Then> and in fact is the most important part of the label <IF>.

<Then>

a. New initialmarking.place ID: this element shows the new initial marking after firing.
b. New.place ID: this element shows the new initial value of α that is obtained after firing.

4.2 Secondary elements of Petri net

Nets can include sub-elements. Different types of Petri nets for secondary elements indicate different concepts. In this section, there is a secondary element and it is the element <page> that includes characteristics such as:
a. id: a unique identity that provides the possibility of reference to net by another net.
b. description: this element is optional and contains description of an element.
c. THEN: this element has one characteristic and it is the characteristic β.
d. New.place ID: this element indicates the value of α and is calculate as follows:

$$New\ \alpha = \alpha.\ P_{source} \times CF(\mu) \tag{1}$$

5. Implementation and evaluation of suggested method

With regard to suggesting a grammar for stating phase colored Petri net in previous section, implementation and evaluation of this suggested method are considered next.

5.1 Case study

In this article, system of bank facilities will be considered and Petri net corresponding to it will be drawn. For description of this net, suggested grammar presented in the previous section and phase behavior of net will be indicated by CPNML Language. Fuzzy rules are as follows:

if p_1 then p_2
if p_4 and p_5 then p_6
if p_2 then p_3
if p_3 and p_6 then p_7
if p_2 and p_6 then p_9
if p_9 then p_{10}
if p_7 and p_{10} then p_8

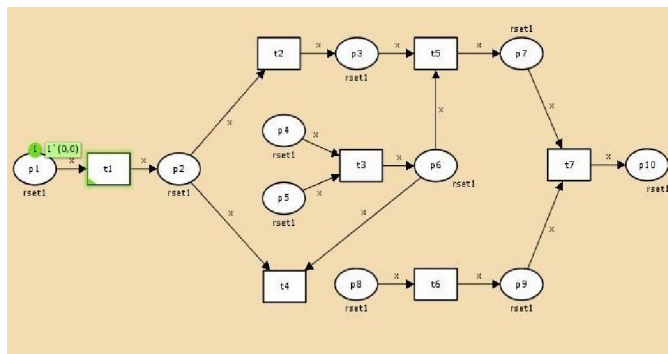**Fig. 1.** Colored Petri net corresponding to system of bank facilities

For example, rule "If p_4 and p_5 then p_6" that has been given in figure means that if (positive) experience of customer's work with bank is high (p_4) and has a high financial flow (p_5), then it does not need to give a valid document (p_6). Grammar that supports concepts of phase colored Petri net are given for place p_1, transition t_1 arc ar_1.

6. Results

In this paper, we have presented a Markup Language that supports fuzzy formal model. We have explained about some activities accomplished on fuzzy XML tags and injecting uncertainty in fuzzy Petri net. In addition, a Markup Language has been presented for supporting fuzzy formal model, writing a fuzzy colored Petri net model into fuzzy markup language.

Acknowledgement

References

Brosso, I., Bressan, G., & Ruggiero, W.V. (2002). Simulation e- business applications using PNML. *In Proceedings of the 2nd WSEAS International Conference on Simulation, Modeling and Optimization* (ICOSMO) (Skiathos, Greece, Sep. 25-28).

Gaurav, A., & Alhajj, R. (2006, April). Incorporating fuzziness in XML and mapping fuzzy relational data into fuzzy XML. In *Proceedings of the 2006 ACM symposium on Applied computing* (pp. 456-460). ACM.

Gehlot, V., & Nigro, C. (2010, December). An introduction to systems modeling and simulation with colored Petri nets. In *Simulation Conference (WSC), Proceedings of the 2010 Winter* (pp. 104-118). IEEE.

Jensen, K., Kristensen, L. M., & Wells, L. (2007). Coloured Petri Nets and CPN Tools for modelling and validation of concurrent systems. *International Journal on Software Tools for Technology Transfer, 9*(3-4), 213-254.

Kindler, E. (2006). The petri net markup language and iso/iec 15909-2: Concepts, status, and future directions. *Entwurf komplexer Automatisierungssysteme, 9*, 35-55.

Ma, Z. M., Liu, J., & Yan, L. (2010). Fuzzy data modeling and algebraic operations in XML. *International Journal of Intelligent Systems, 25*(9), 925-947.

Zhang, F., Yan, L., Ma, Z. M., & Cheng, J. (2011, March). Knowledge representation and reasoning of xml with ontology. In *Proceedings of the 2011 ACM Symposium on Applied Computing* (pp. 1705-1710). ACM.

Zhang, F., Ma, Z. M., & Yan, L. (2012, June). A fuzzy ontology approach for representing fuzzy petri nets. In *Fuzzy Systems (FUZZ-IEEE), 2012 IEEE International Conference on* (pp. 1-8). IEEE.