

Performance evaluation of enterprise architecture with a formal fuzzy model (FPN)

Ashkan Marahel^{a*}, Ali Harounabadi^b and Seyyed Javad Mirabedini^b

^aDepartment of Computer Engineering , Science and Research Branch , Islamic Azad University, Boushehr, Iran

^bFaculty Member of Computer Science , Central Branch , Islamic Azad University, Tehran, Iran

ARTICLE INFO

Article history:

Received March 29, 2012

Received in Revised form

June, 18, 2012

Accepted 29 July 2012

Available online

August 3 2012

Keywords:

Fuzzy UML

Uncertain requirements

C4ISR

Fuzzy Petri Net (FPN)

ABSTRACT

Preparing enterprise architecture is complicated procedure, which uses framework as structure regularity and style as the behavior director for controlling complexity. As in architecture behavior, precedence over structure, for better diagnosis of a behavior than other behaviors, there is a need to evaluate the architecture performance. Enterprise architecture cannot be organized without the benefit of the logical structure. Framework provides a logical structure for classifying architectural output. Among the common architectural framework, the C4ISR is one of the most appropriate frameworks because of the methodology of its production and the level of aggregation capability and minor revisions. C4ISR framework, in three views and by using some documents called product, describes the architecture. In this paper, for developing the systems, there are always uncertainties in information systems and we may use new version of UML called FUZZY-UML, which includes structure and behavior of the system. The proposed model of this paper also uses Fuzzy Petri nets to analyze the developed system.

© 2012 Growing Science Ltd. All rights reserved.

1. Introduction

During the past few years, companies and organizations are working hard to provide better services. This leads organizations to have better processes (Lindsay et al., 2003) and information technology as well as the change agents in organizations (Daft, 2000). For any production, these issues force organizations look for essential information technology, appropriate, correct and valuable use of IT in usage of enterprise IT architecture or enterprise architecture. Architecture consists of a large number of documents, which describe each part of the organization. The problem is to know how to apply all requirements. To regularize and to organize architectural description, we need necessary framework. Enterprise architecture's framework is intended to focus on one aspect of the organization without losing the holistic approach for all stakeholders. There are different frameworks for enterprise architecture but in this article, the C4ISR architecture's framework is considered. This is because of

* Corresponding author. Tel:+989166317348

E-mail addresses: ashkan.marahel@yahoo.com (A. Marahel)

its methodology of production in the framework and also features a component-level aggregation and minor revisions. The lack of a unified modeling language in a structured approach causes the reduction on readability and efficiency of many architecture methods. The problem in object-oriented methods rises by presenting unified modeling language (UML).

UML language as a powerful language can support object-oriented concepts and it has been widely implemented in different well-known databases. Techniques used in UML can evaluate certain cases but there is uncertainty in many information systems. By applying uncertainty in UML, new version of UML called FUZZY-UML represents better concept of this language. According to Behbahaninejad et al. (2012), UML has been recognized as one of the most popular techniques to describe static and dynamic aspects of software systems. One of the primary issues in designing software packages is the existence of uncertainty associated with such models. Fuzzy-UML to describe software architecture has both static and dynamic perspective, simultaneously. The evaluation of software architecture design phase initiates always help us find some additional requirements, which helps reduce cost of design. They used a fuzzy data model to describe the static aspects of software architecture and the fuzzy sequence diagram to illustrate the dynamic aspects of software architecture. They also transformed these diagrams into Petri Nets and evaluated reliability of the architecture.

The products of enterprise architecture framework, C4ISR, are described and illustrated by UML diagrams. So far, many methods have been proposed for the assessment of enterprise architecture by converting C4ISR products shown with UML diagrams to Petri networks to evaluate enterprise architecture (Shin et al., 2010; BAI et al., 2008). According to Afshani et al. (2012), C4ISR framework describes architecture in three views of information architecture and defines a set of products, which are main outputs of enterprise architecture design. Formats and templates, which are presented for C4ISR products, cannot describe uncertainty in process or data. Meanwhile, uncertainty in many information systems is unavoidable and using the concept of fuzzy numbers in architecture design helps architects handle uncertainty in process and data of organization. Afshani et al. (2012) presented a new template based on fuzzy-UML concept for some of C4ISR products such as Logical Data Model (OV-7), Operational Event/Trace Description (OV-6c) and Systems Event/Trace Description (SV-10c). In this article, we plan to do an assessment of enterprise architecture in fuzzy form. The products of C4ISR enterprise architecture are classified into three categories: operational, systemic and technical and in this paper, for evaluating the performance of enterprise architecture we focus more on systems products such as SV-4, SV-10 and SV-11. The proposed idea in this article is considered in Fig. 1.

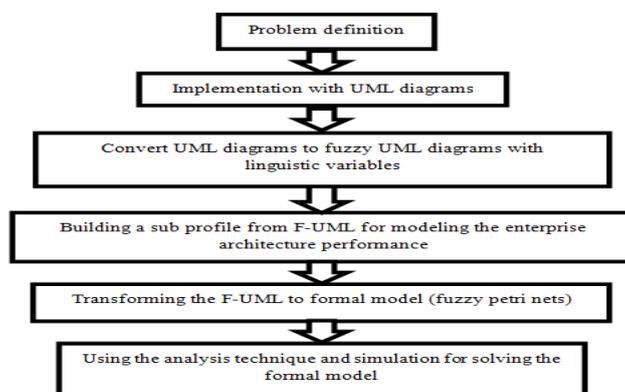


Fig. 1. Presented model of this study

The second part of this article evaluates the works, which have done until now in enterprise architecture and the third section of the paper deals with fuzzy data. This section presents a class diagram with a fuzzy numbers and uses uncertainty in the data structure and then by using diagrams for sequence cases mode and activity as fuzzy, we perform modeling fuzzy behavior. In forth section of this paper, diagrams obtained based on section, some algorithms are converted to colored fuzzy Petri nets. The fifth section deals with an experiment on the system and, finally, section six concludes and offers work for the proposed work.

2. Related works

Rezaei and Shams (2009) described the ways of evaluating the process of enterprise architecture. Javanbakht et al. (2008) proposed a method for optimal design architecture. One of the methods used for breeding and development of enterprise architecture is assessment of enterprise architecture maturity. The problem is that architectures in different organizations normally do not have enough talent for improvement. In Javanbakht et al. (2008)'s paper, a method to measure whether they have enough talent to improve was presented. Javadpour and Shams (2009) presented a model for evaluating the performance of enterprise structure. They first converted products of C4ISR framework enterprise structure into UML diagrams and then by proposing an algorithm, they converted the UML diagrams into Petri net graphs and finally by using the Petri nets, they measured the performance of enterprise architecture. Mozafari et al. (2011) first described a way to verify the behavior of enterprise architecture and then by using an algorithm they converted the products of enterprise architecture into Petri Nets diagrams and evaluated them. All works cited by the researchers focused on certain aspects of systems but when we consider uncertainty in models, most of these algorithms become inefficient. What distinguishes this paper from other papers is the focus on uncertain aspects of system that fuzzy logic used for this work.

3. Modeling fuzzy data and C4ISR for evaluating performance by linguistic variables

As we explained in the previous section, the framework of C4ISR enterprise architecture has lot of products but for performance evaluation needs to investigate some of its products. These products include: SV-4, SV-10, SV-11. Each of these products for their expression use one or more of the UML diagrams to model the product SV-4 (describing function of system). We use the activity diagram for product SV-10 state and sequence diagram and for SV-11 (physical data model) the class diagram is used. Therefore, we first must convert these diagrams to fuzzy diagrams by using linguistic variables.

3.1. Data model of fuzzy UML

For modeling fuzzy data, the UML diagrams are used. Class diagram in UML are logical model, which describes the main structure for the system. The classes and relationships among them are elements of class diagram. By entering the uncertainty to these elements, FUZZY-UML data model is created. Based on data extracted there are three levels of the fuzzy system as follows: making fuzzy structure within class, the data model apply according to its contents in this section as fuzzy characters. The question is whether the proposed fuzzy modeling operates efficiently or not.

Even when we assume a class structure is certain, and some objects with dependent level belong to the class $[0,1]$ the third level of fuzzy modeling can be applied as class subsection levels. An attribute in a class is described on a assumed range for its slope attribute or class name must express with the

words “ WITH MEM DEGREE” where $0 \leq MEM \leq 1$. A membership degree in which this attribute belongs to the class or class belongs to data is given. In the third level, a key word fuzzy appears in front of the attribute. In the second level on fuzzy modeling, we should represent the degree of dependence of one class sample to the class. For this purpose, an additional attribute in the class is defined to show the degree in which the sample belongs to the class and its slope is $[0,1]$ denoted by μ . Ma (2011) provided an example about the bank account that fuzzy concepts are used in it. In Class shown in Fig. 1, credit can be adjective consists of fuzzy values (third level of fuzzy modeling). In other words, credit adjective is a language variable where its scope includes fuzzy sets (e.g. low/high). Type of class represents membership degree of credit adjective to the class (the first level of fuzzy modeling) :

“ credit with 0.8 membership DEGREE “

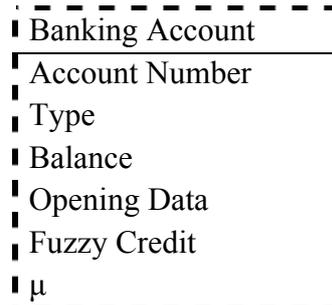


Fig. 2. Fuzzy class diagram of bank account

3.2. Fuzzy use case

When a service is presented as definitive, fuzzy use case becomes clear. After determining the definitive and indecisive cases, we draw the diagram of use case in UML, for applying use cases, we use sequence diagrams. If the use case is uncertain, the sequence diagram is uncertain too. A sequence diagram includes the operating, system and the message, where each message will include events and circumstances for the event. This diagram uses fuzzy rules for transforming from one object mode to another that fuzzy rules can be written as follow:

RULE= if <condition list> then <event list>

According to Haroonabadi and Teshnehlab (2008), uncertainty in method including two-stage fuzzy modeling has two stages, in the first stage the method membership degree is chosen and in the second stage decision logic is determined. Fig. 2 shows a sample fuzzy concept where an object with membership function $\mu_B(A)$ transforms message of C to B object, and B object with membership function $\mu_A(x)$ transforms message of D to A object.

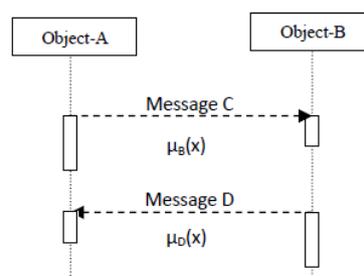


Fig. 3. Fuzzy Sequence Diagram

3.3. Activity diagram

Use case diagram do not show orders of implementation of use cases. To achieve this grand, activity diagram can be used to display system activity, the following rules are used to make fuzzy activity's diagram:

1) The rules of simple fuzzy extraction:

$$R: \text{if } a \text{ then } c \text{ (cf}=\mu) \text{ , } \square, w. \tag{1}$$

These kinds of fuzzy rules by using activity diagram are shown in Fig. 4,

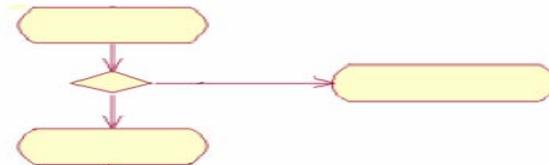


Fig. 4. Activity diagram with rules of simple fuzzy

2) Combination references rule:

$$R: \text{if } a_1 \text{ AND } a_2 \text{ AND } \dots \text{ AND } a_n \text{ then } C \text{ (cf}=\mu) \lambda, w_1, w_2, \dots, w_n. \tag{2}$$

To draw this type of fuzzy rules in activity diagram, we perform as shown in Fig. 5.

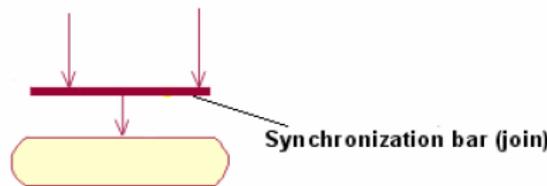


Fig. 5. Activity diagram with combination reference rule

3) Combination of seasonal rules:

$$R: \text{if } a_1 \text{ OR } a_2 \text{ OR } \dots \text{ OR } a_n \text{ then } C \text{ (cf}=\mu) \lambda_1, \lambda_2, \dots, \lambda_n, w_1, w_2, \dots, w_n. \tag{3}$$

To draw this type of fuzzy rules in activity diagram, we perform as shown in Fig. 6.



Fig. 6. Activity diagram with combination seasonal rule

3.4. State diagram

State diagram models different modes of object. These graphs show the behavior of dynamic systems and Fig. 7 shows an example of a state diagram:

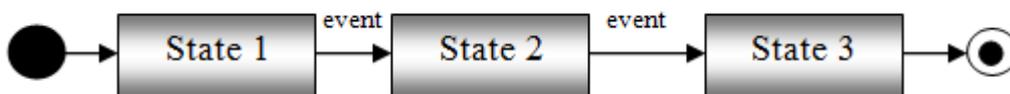


Fig. 7. State Diagram

A fuzzy state diagram is a graphical model in fuzzy UML, which shows different modes of a fuzzy object in real world. This diagram uses the fuzzy rules for transforming one object's mode to another object. A fuzzy rule is shown as follow:

<on event list<event threshold>>
 If condition list <EC coupling>
 Then action

Each part of state diagram can be converted to the fuzzy state diagram and the following table clearly shows this conversions:

Table 1
 Convert state diagram to fuzzy state diagram

State diagram	Fuzzy state diagram
 state [condition] Event	 Action of rule [fuzzy condition] Fuzzy event

4. Fuzzy Petri Net

According to Motameni et al. (2008) fuzzy Petri networks can be applied to model the fuzzy rules defined as follow,

($P, P_s, P_e, T, TF, TRTF, A, I, O, TT, TTF, AEF, PR, PPM, TV$)

P is a finite set of places. Each location is expressed by the following features:

$P_s \subset P$: a finite set of input places for primary events.

$P_e \subset P$: a set of output places for operations or the results.

T : is a finite set of fuzzy transitions. This places use the values of input places and produce values for output places.

TF : is a finite set of transition functions.

$TRTF: T \rightarrow TF$ is a transition function that mapped every membership T to a member function of TF .

$A \subset (P * T \cup T * P)$: is a finite set of connections between places and transitions. The connections between input and transition sites ($P*T$) and the connections between transitions and output places ($T*P$) are expressed by them. So:

$I: P \rightarrow T$ is an input mapping.

$O: T \rightarrow P$ is an output mapping.

TT : is a finite set of fuzzy token type. Each token is linguistic variable (eg “low , high or medium “), which are expressed by a membership function.

$TTF: T \rightarrow P$ is the function of token type that mapping each fuzzy place of member P to fuzzy token type of member TT .

$AEF: Arc \rightarrow Expression$ is a phrase function that mapping each connection to a phrase that holds the information.

PR : is a finite set of transitions that operate correspond to an event , condition , or operations.

$PPM: P \rightarrow [0,1]$ is a fuzzy location for mapping transition where: $|pr|=|p|$.

$TV: P \rightarrow [0,1]$ the correct amount of tokens that mapping to table to the locations and express membership degree of a token to special location.

5. Fuzzy UML conversion algorithm to fuzzy Petri Nets

Due to the inability of UML in evaluation software systems, the necessary of converting real model (UML) to the formal model (FUZZY Petri networks) is observed. Note that in this article for expression one of the products (SV-10) we used fuzzy sequence diagram, here we give an algorithm that converts this diagram into colored Petri networks (Bostan-Korpeoglu & Yazici, 2006; Hájek & Olej, 2008). The algorithm is as follow:

Step 1) For each message in sequence diagram, all the events and circumstances should be specified. The events and conditions calculated for the dream activity is represented in Table 1.

Table 2

Events and condition for dream activity

Rule	Event	condition
R1	E is E ₁	C ₁
R2	E is E ₂	C ₂

Step 2) The right conditions must be investigated. Therefore, we need to map it. For each condition, we should put a transition, which is responsible for validation of results. This means that the token can be expressed with a fuzzy value and its value is between 0 and 1. The complete analysis of the condition c, which would be a fuzzy value and it depends on condition. However, most of the times, we face with more than one condition. These conditions can be expressed as follows,

$$OR = \mu_{AB}(x) = Max[\mu_A(x), \mu_B(x)] \tag{4}$$

$$AND = \mu_{AB}(x) = Min[\mu_A(x), \mu_B(x)] \tag{5}$$

In Fig. 8 and Fig. 9, two different common cases are depicted.

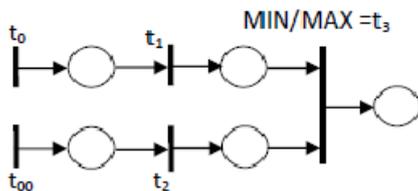


Fig. 8. Facing with more than one condition (OR)

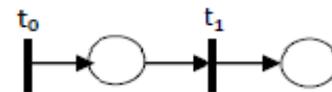


Fig. 9. Facing with more than one condition (AND)

Step 3) events run if conditions are established. Fuzzy value calculated for evaluation results are used. Here, to apply correct previous amount, the result membership function is used and Fig. 10 shows aggregation.

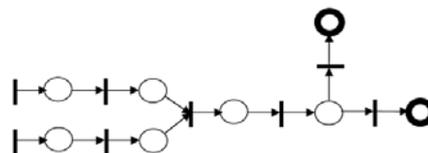


Fig. 10. Modeling consequent of fuzzy rule with aggregation

Finally, as the final output of a fuzzy system should be a certain value, we must convert the fuzzy value to a certain amount. This operation is based on COG test is done as follow:

$$COG = \int_a^b \mu_A(x)dx / \int_a^b \mu(A)dx \tag{6}$$

6. Case study

In this section, we evaluate the performance of an organization using the case study given in Motameni et al. (2008). This is the organization for describing climate. Suppose that this organization wants to use an online system. For expressing climate, people enter to this and it reports the weather based on weather conditions announced by the meteorological agency. Since expressed weather conditions are uncertain (e.g., air pressure up or down), the FUZZY UML is used. According to the products listed in C4ISR framework, for expressing SV-10 products we use sequence diagram in our system, which is non-deterministic system. For describing the performance of this system, we use the following fuzzy sequence diagram:

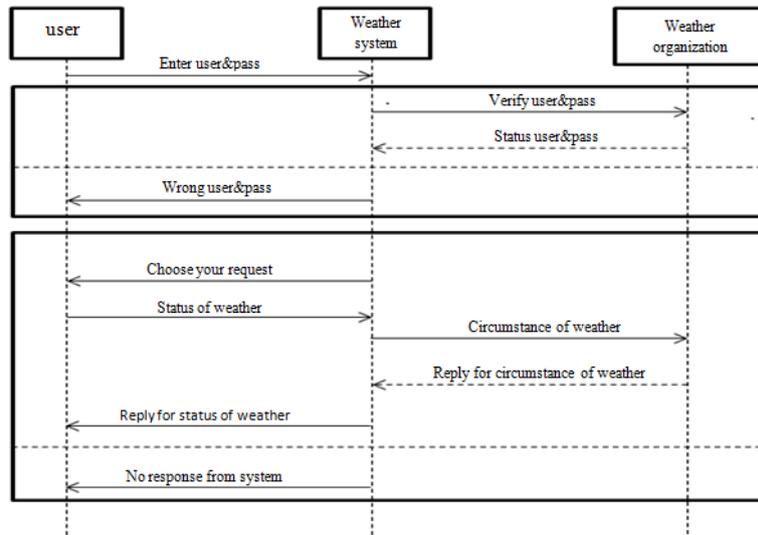


Fig. 11. Fuzzy sequence diagram for Weather Systems Online

For example, linguistic variables of temperature (high , medium , low) convert to crisp value by using membership function as Fig. 12 and Table 3 shows FUZZY rules' table of weather system.

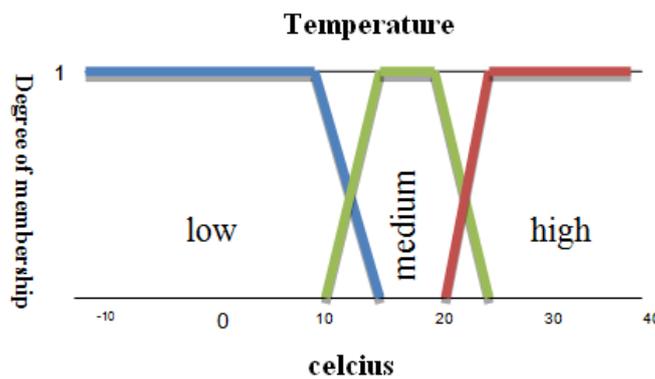


Fig. 12. Membership function for temperature

Table 3
Fuzzy rules for weather system online

Rule	Condition	Table state
R1	If(temperature is high & pressure is low &cloudy is unstable & humidity is medium & windy is strong)	Weather is storm
R2	If(temperature is low & pressure is high &humidity is high or humidity is medium)	Weather is Hail
R3	if(temperature is medium & cloudy is unstable &windy is strong or moderate)	Weather is rainy
R4	If(cloudy is stable & humidity is high &windy is calm)	Weather is snow
R5	If(pressure is low &humidity is low & windy is calm)	Weather is fog
R6	If(temperature is medium & pressure is low & humidity is high or medium)	Weather is shower

After drawing fuzzy sequence diagram of weather online systems, according to the proposed algorithm, we convert it to colored fuzzy Petri Nets. The obtained diagram is shown in Fig. 13. In this section, we evaluate the performing model from the stakeholders' perspective in architecture. For simulation Petri diagrams, the software CPN/TOOLS and SHOW FLOW used.

Response time:

Response time is equal to the time where person or agent spends to receive service in the organization uses and this time includes the waiting time in queue and the time spent for receiving time service.

Processing time:

Processing time is equal to the time interval between the beginning and the end of a process for the specific input. This time equal to the total time spent waiting in queue and the time spend for receiving services.

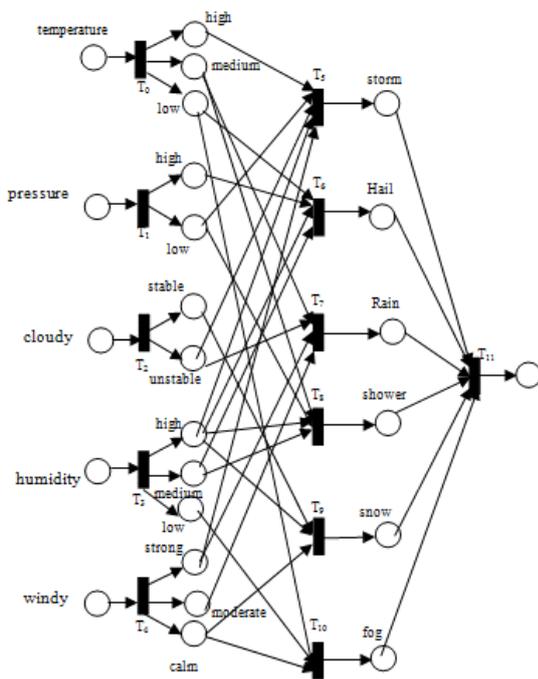


Fig.13. Fuzzy Petri nets for weather system online

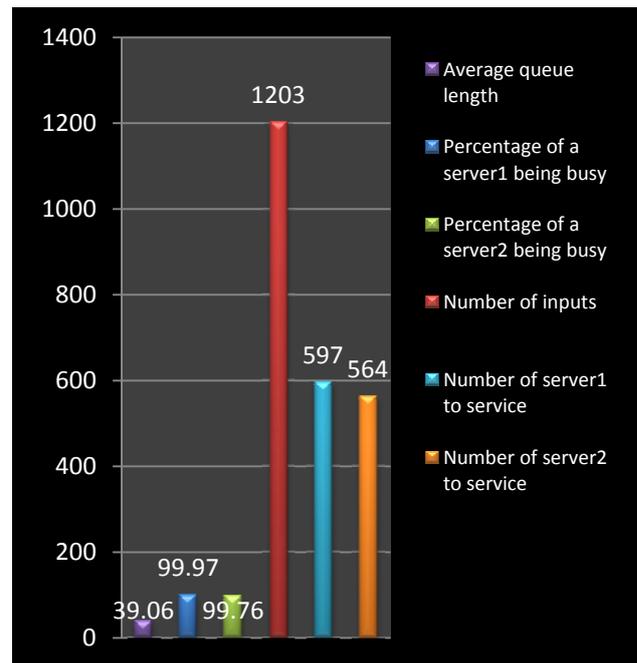


Fig. 14. Performance parameters diagram

Queue length:

Queue length is equal to the number of vertebrae in the queue whenever the queue is removed from the vertebral or nut to be put in the queue.

Performance amount:

Operational efficiency is the percentage of time that a resource is busy. Given the above description, we assume an exponential distribution with parameter time entry system is 0.1, Also, if the service time of each server to the client is an exponential distribution with parameter 0.5 and the second server is in the existing system and also out of people is an exponential distribution with parameter 0.3. Due to the mentioned parameters, the average queue length, the average percentage of busy server 1 and server 2, the number of inputs to the system and also the number of server to server 1 and 2 is plotted in the graph below:

7. Conclusion

In this paper, we have presented an algorithm for evaluating the performance of C4ISR enterprise architecture framework's products to do this. At first by using FUZZY-UML charts, we modeled products for performance evaluation. However, since UML is disabled in the evaluation of software systems, we transformed these charts to fuzzy Petri networks by using an algorithm. Since the most requirements of real systems are inconsistent, in our proposed model to cover this issue, we used the fuzzy logic. Considering that in most software systems, certain parameters have been used to evaluate performance. In this article, we have tried to examine almost all performance parameters from perspective of system's stakeholders.

References

- Afshani, J., & Harounabadi, A., & Abbasi Dezfouli, M. (2012). A new method for designing uncertain enterprise architecture. *Management science letters*, 2, 689–696.
- Bai, X.M. (2008). An application with UML object-based PETRI NETS for C4ISR architecture simulation validation. *Proceeding of a seventh international conference on machine learning and cybernetics*, Kunming.
- Behbahaninejad, P., Harounabadi, A., & Mirabedini, S.J. (2012). Evaluating software architecture using fuzzy formal models. *Management Science Letters*, 2, 469–476.
- Bostan-Korpeoglu, B., & Yazici, A. (2006). A fuzzy Petri Net model for intelligent database. *Data & Knowledge Engineering*, 8, 112-122.
- Deft, R. (2000). *Organization Theory and Design*.
- Javanbakht, M., Rezaie, R., Shams, F., & Seyyedi, M.A. (2008). A new method for decision making and planning in enterprises. *Proceedings of the 3rd International Conference on Information and Communication Technologies: From Theory to Applications (ICTTA 2008)*, 7-11.
- Javadpour, R., & Shams, F. (2009). *Performance evaluation of electronic city architecture using colored Petrinets*, The 2nd Conference on Electronic City, Tehran.
- Lindsay, A., Downs, D., & Lunn, K. (2003). Business processes –attempts to find a definition. *Information and Software Technology*, 45, 1015-1019.
- Haronabadi, A., & Teshnehab, M. (2008). A novel method for behavior modeling in uncertain information systems. *World Academy of Science, Engineering and Technology*, 41, 959-966.
- Hájek, P., & Olej, V. (2008). Air quality modeling by Kohonen's self-organizing feature maps and LVQ neural networks. *WSEAS Transaction On Environmet And Development*, 1(4), 45-55.
- Ma, Z. (2005). Fuzzy information modeling with the UML. *Idea Group Publishing*, 153-176.
- Ma, Z.M., Zhang, F., & Yan, L. (2011a). Fuzzy information modeling in UML class diagram and relational database models. *Applied Soft Computing*, 11, 4236-4245.
- Ma, Z.M., Yan, L., & Zhang, F. (2011c). Modeling fuzzy information in UML class diagrams and object oriented database models. *Fuzzy Sets & Systems*, 186, 26-46.
- Motameni, H., Movaghar, A., Daneefar, I., Nematzadeh, H., & Bakhshi, J. (2008). Mapping to convert activity diagram in fuzzy UML to fuzzy Petri Net. *World Applied Sciences Journal*, 3(3), 514-521.
- Mozaffari, M., Harounabadi, A., & Mirabedini, S.J. (2011). A method for validating a behavior of enterprise architecture. *World Applied Sciences*, 14(6), 831-841.
- Rezaei, R., & Shams, F. (2009). Providing a comprehensive method for developing and evaluating enterprise architecture plan, *The first Conference on Enterprise Architecture in Practice*, Isfahan, Iran.
- Shin, M.E., Levis, A.H., & Wagenhals, L.W. (2010). Transformation of UML-based system model to design/CPN model for validating system behavior. *System Engineering*, 5(4), 288-312.
- Zadeh, L. A. (1983). The role of fuzzy logic in the management of uncertainty in expert system. *Fuzzy Sets Systems*, 11, 199-227.