A Well-Formed Petri Net Based Modeling Approach for Emergency Situation handling in Hospital

Suparna Karmakar
Assistant Professor
Department of Information Technology
GNIT, Kolkata, WB, INDIA

Sayani Chandra
Assistant Professor
Computer Science and Engineering Department
GNIT, Kolkata, WB, INDIA

Abstract- This work propose a petri net model based approach for emergency situation handling in a Hospital area. In a hospital it is always need to maintain quiet and satisfactory situation for patients. But often some unwanted emergency situation occurs may be due to natural calamity or accidentally. To handle this type of scenario a petri net based model has been proposed . In this investigation the Well-Formed PetriNet models have been used to construct the emergency situation handling. Finally the structural reachability also has been examined.

Keywords – PetriNet , Well-Formed PetriNet(WFPN) , Rechability, Liveness

I.INTRODUCTION

Emergency situation handling in a hospital is a vital issue for patients' safety. In a hospital there may be various unwanted situation like fire etc. When this type of unwanted situation occurs then there must have some monitoring system so that proper handling is possible. To establish a robust framework for such type of information processing system need a good mechanism for modeling. The current trend in high speed and wide bandwidth internet technologies provides us with tremendous amount of information processing in an immediate manner. Petri net is one of the modeling technique which can be used for depicting various information system having concurrent, asynchronous activity and its mathematical analysis technique can be used for formal verification of the system whether the system is deadlock free or not.
In this work, a subclass of Petri Net namely Well-Formed Petri Net(WFPN) has been used to model the emergency situation handling scenario.

II. BASIC CONCEPT

A. PETRI NET

Petri nets were developed in the early 1960s by Carl Adam Petri [1]. Later on, it was widely generalized and investigated for further improvement in modeling. From [5], [7], [8] it can be found that how achievements in modeling method of Petri Net has been used in various fields such as computer science, automation and computer integration manufacture.. The formalized definition can be found in [2], [3]. Petri nets are a graphical and mathematical modeling tool for describing and studying information processing systems that are characterized as being concurrent, asynchronous, parallel, nondeterministic and/or stochastic. As a graphic instrument, it owns a graphically depicting function and simulates the dynamic behaviors of the system through the flow of tokens. And as a mathematical tool, it depicts system behaviors through building state equation (analysis of dynamic behavior can be found in [1], [4]). Petri nets are defined by a quadruple (S, T, F, W), where

1. S is a finite set of places, represented by circle or ovals.
2. T is a finite set of transitions, represented by squares or rectangles.
3. F $\subseteq$ {S x T} U {T x S} is the flow relation. Arcs connect places and transitions. Inward arcs go from a place to a transition and an outward arc goes from a transition to a place.
4. W: F$\rightarrow$N-{0} is the weight function, which associates a nonzero natural value to each element of F. If no weight value is explicitly associated with a flow element, the default value 1 is assumed for the function.

Tokens reside in the places. They are represented by a solid circle or by a dot inside of a place. The execution of Petri net is controlled by the position and movement of tokens. A Petri net is a kind of directed graph with initial marking M0. A marking (state) assigns to each place a positive integer. If a marking assigns a non-negative integer k to a place s, then s is said to be marked with k tokens. A marking is denoted by M, a m-vector. The s^{th} component is denoted by M(s), which represents the number of tokens at place s [1].

B. TRANSITION ENABLING RULE

A transition t is said to be enabled if each input place s of t is marked with at least w(s, t) tokens, where w(s, t) is the weight of the arc from s to t. For a given marking the firing of an enabled transition results in a successor marking M', Where,

$$M(s) = M(s) – w(s, t) + w(t, s) \qquad \text{.....................(1)}$$

w(s, t) is the weight of the arc between s and t. w(t, s) is the weight of the arc between t and s.

C.COLOR PETRINET:

There are several extension of PetriNet. Colored PetriNet (CPN)[11] allows us with multiple types, called *colors*, attached to tokens where the transition arises only if identical types of tokens get ready. CPN allows us with a notion of identity reference on each token so that special kinds of constraints are posed on places and transitions. Some CPN contains transitions where input-tokens should be kept for delay time until completion. Such kind of CPN is called timed CPN (or CTPN). The procedure of designing the hierarchical model can be found in [10].

There are three fundamental issues here, boundedness, structural liveness and reachability. Boundedness issue means we should examine whether the number of tokens at every place increases infinitely or not, liveness issue whether all the transitions can be fired or not, (i.e., whether there exists a ”deadlock” situation), and reachability issue whether any tokens get to a place of interests eventually from initial positioning. Among others, the reachability problem is known to be decidable in basic PetriNet but computationally expensive, and sometimes undecidable[6].

In this investigation we assume one source and one sink, so reachability means every token comes from the source into the sink which corresponds to ”normal” execution. It takes exponential time to examine reachability in a naive manner because we should exhaust all the places and transitions, even if no loop exists. In a case of CPN, we should have color constraints at each place and transition. To examine reachability, there are two aspects to be discussed, structural reachability and liveness. The structural reachability concerns connectivity, i.e., whether there exists a path to a node from the source or not. The liveness means we should examine whether every node gets any tokens or not in finite times. Because model components concern undeterministic and asynchronous behaviors, the problems would become undecidable without any constraints. From the algebraic point of view, reachability may not hold if one token goes in an undeterministic way at one place, if token goes in a repeated manner with an arc coming back to a node, or if a node means a function which invokes functions in a nested way. That's why we need well-formed PetriNet.

D. WELL-FORMED PETRI NET:

WFPN[9] is a subclass of petri net where we assume nodes and transitions (of n-in/m-out) and arcs between them. A PetriNet frame y = α(x) is defined as two nodes and one transition connected together as transition T from x to y (thru T), shown in figure 1 (a). We also assume one source node (”source”) and one sink node (”sink”) in advance.

Given two nodes x and y, let us define Well Formed PetriNet (WFPN), y = E(x). In the following, X, Y mean variables over nodes, a label _ and n > 0. Any nodes x, y may have labels A,B, such as B : y =E(A : x). Now we define following six constructs of TRANS, SEQ, DISJ, ASYNC, LOOP, LABEL starting from a normal WFPN sink =α(source).

(1) Transition : Both Y = α(X), Y = $\ell(X)$are WFPNs. Note _ plays role of transition but a function call practically.

(2) Sequence : Let Y1 = E1(X1), Y2 = E2(X2) be two WFPNs, then a sequence of Y = E2(E1(X)) is also a WFPN denoted as Y = (E1 ・ E2)(X) in figure1(b).

(3)Disjunction : Let Yi = Ei(Xi) be a WFPN, i = 1, .., n, and p1+...+pn = 1 where 0 < pi ≤ 1. Then Y = (E1/p1 + ... + En/pn)(X) from X to Y is also a WFPN, where one of the WFPNs Y = Ei(X) is selected with the probability pi.The WFPN is also called probabilistic as shown in figure 1 (c). If all pi are equal values (= 1/n),we may omit them.

(4) Asynchronization : Let $Y_i = E_i(X_i)$ be a WFPN, $n > 0$, $i = 1, .., n$. Then a bundle of $Y = [E_1, ..., E_n](X)$ from X to Y is also a WFPN where all the $E_i(X)$ be combined into Y shown in figure 1 (d). The WFPN is called *asynchronous*.

(5) Loop : Let Y = E(X) be a WFPN and U a parameter. Then Y = [U] : E(X) is a WFPN in figure1(e). That is, we repeat Y = E(X) process while U is true.

(6)Label : Let Y = E(X) be a WFPN, ℓ a new label and U a parameter. Then Y =$\ell[U]$: E(X) is WFPN, called declaration as in figure 1 (f).The label ℓ plays a role of a function name given to E, and may ppear as a label of transition within E with an argument to U.

(7) Normal WFPN : No other than (1)-(6) is a WFPN. Afterwards we discuss a normal WFPN, sink=E(sourceX), where E is a WFPN.

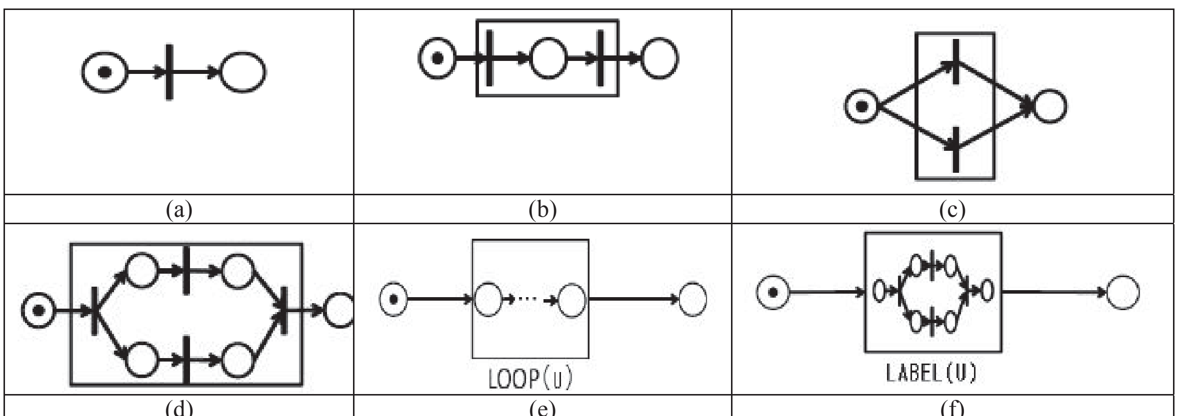

Figure1: Constructing WFPN

Assume X, Y mean any node. First of all, let us define TRANS construct corresponding to Y =α(X), Y = _(X) as shown in figure 2(a). We apply TRANS construct to obtain Y = _(X) as a label _ (corresponding to an WFPN) with an input X and an output Y . A special form of TRANS is given in advance in any case with ”source” and ”sink” corresponded to a normal WFPN. A construct SEQ takes two WFPN models Y = E1(X), V = E2(U) as input and build V = E1・E2(X) as shown in figure 2(b). Note an intermediate new node is generated to connect both WFPNs. Two constructs DISJ and ASYNC describe disjunction and asynchronization respectively. Each takes n WFPNs, Y1 = E1(X), ..., Yn = En(Xn) as inputs and build Y = (E1 + ... + En)(X) and Y = [E1, ...,En](X) respectively as shown in figure 2(c) and (d). In DISJ,no node nor transition is generated but path connections be changed. On the other hand, in ASYNC, two transition symbols are generated at the same time, one for decomposition and another for composition. LABEL and LOOP allow us to construct block strucures. LABEL takes one WFPN *E*, a new label ℓ and a condition *U* as input. The block consists of a WFPN and may contain a transition of the label ℓ within. Such a recursive structure requires well-defined fixed-point semantics and enriches descriptive power.Here we assume U is a meaningful logical formula. In LABEL, no node nor no transition is generated but the label ℓ and the condition

U be attached to the WFPN. LOOP can be seen as LABEL without labels, Y = [U] : E(X), and the condition U be attached to the WFPN E. Starting with a normal WFPN "sink = α(source)", we apply these constructs successively to obtain WFPN models.

III. PROPOSED MODEL

The emergency situation handling approach in hospital consist of basically the following three process: Firstly, Monitoring the emergency situation (denoted as Transition T1), Secondly, action taken for responding to the emergency situation (denoted as Transition T2) and finally Restoring the Emergency(denoted as Transition T3). These three activities must be done sequentially. Figure2 is describing the whole process.

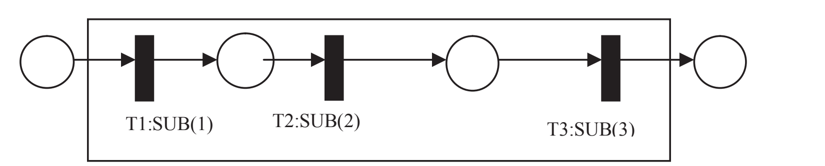

Figure 2:Main process of emergency situation handling

Table -1: Name of the transition with its significance

Name of the transition	Significance of the transition	Name of the transition	Significance of the transition
T1	Preparing for Emergency	T31	Compensating the victim
T2	Responding to Emergency	T32	Releasing the information
T3	Restoring the Emergency	T33	Evaluating the events
T11	Monitoring the Emergency events	T34	Performing feedback
T12	Analyzing the monitoring info	T111	There can be an exhaustive fire in the hospital
T13	preparing for the emergency scheme	T112	A terrorist attack can happen in the Hospital
T14	Create any emergency events and prepare for that	T113	Check whether flood in the area of the hospital
T15	Train maintainer for any emergency events	T114	Check whether there are sufficient storage of medicines and the emergency things.
T16	Involve good emergency event analyzer to prevent the events	T331	Evaluating the events which is the main cause of creating the events
T21	Deciding on the emergency Action	T332	It has to be analyzed that whether any event put not into action
T22	Dispatching the Resources	T131	Keep uninterrupted emergency electric supply
T23	Succoring the Spot	T331	Evaluating the events which is the main cause of creating the events
T24	Reduce the emergency events as soon as possible	T132	Keep back up of emergency medicine.

Now we are elaborating here the sub process of the main three process. In table1 name of all the transitions and their significance has been mentioned. In figure 3 the subprocess of T1 has been modeled which contains another sub function (namely SUB(4)) depicted in figure 4 .Similarly in figure 5 the sub process of transition 2 (T2) has been modeled. In figure 6 the sub process of transition T3 has been designed which also contain another sub process SUB(5) designed in figure 7.

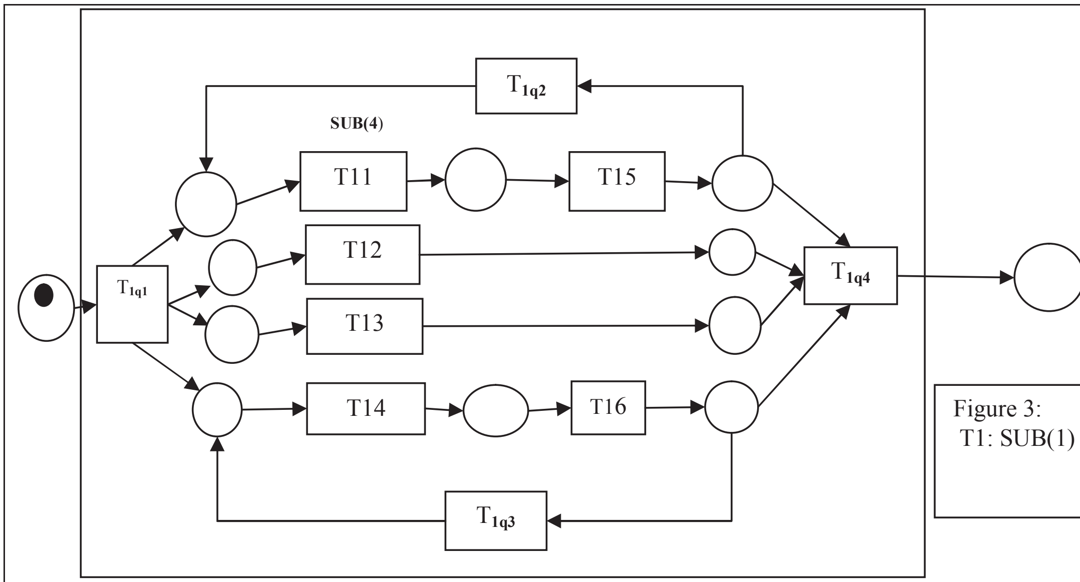

Figure 3:
T1: SUB(1)

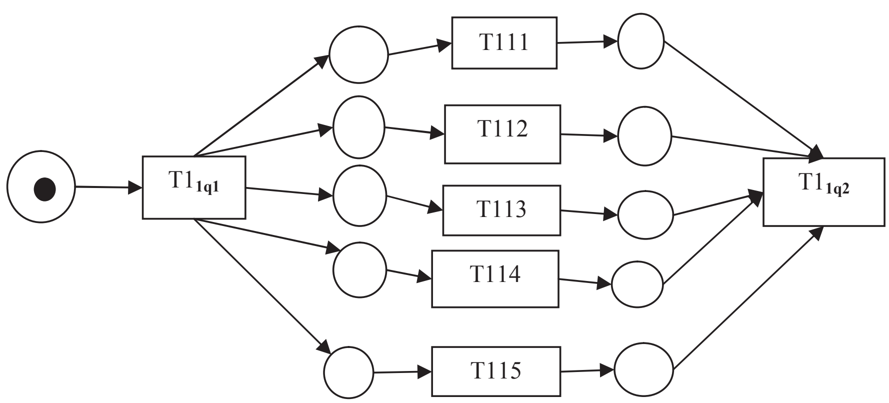

Figure4 :
T11: SUB(4)

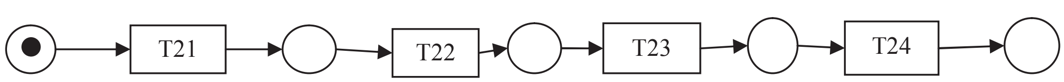

Figure5 : T2: SUB(2)

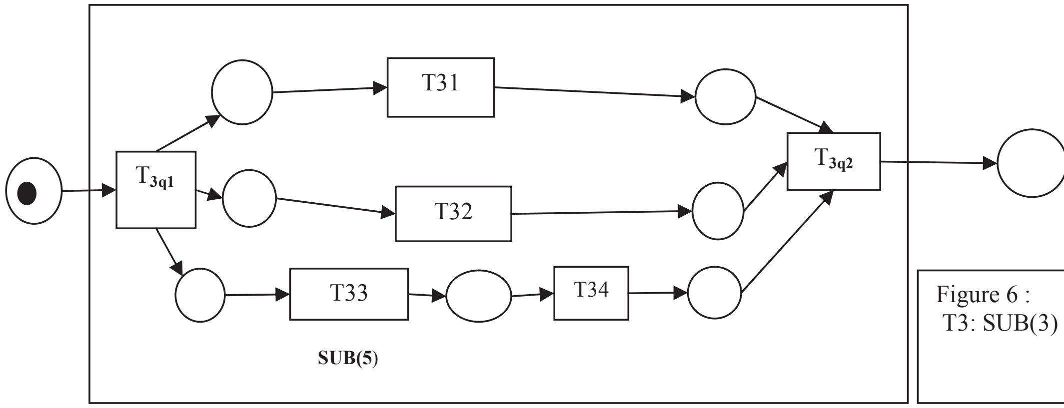

Figure 6 :
T3: SUB(3)

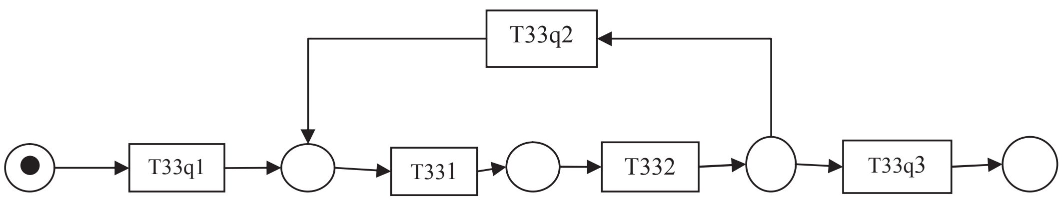

Figure 7: T33: SUB(5)

V.CONCLUSION

In this work a petri net model has been designed and analyzed for handling emergency situation in a hospital. For designing the well-formed petrinet model which is a subclass of petri net model has been used. In this work the structural reachability of the model has been analyzed. As we know that the structural reachability is concerned with connectivity. Here in this work connection between source and sink node has been maintained properly in whole diagram.

REFERENCES

[1] Tadao Murata ― Petri Nets: Properties, Analysis and Application, Proceedings of the IEEE, Vol.77, No.4, April 1989

[2] Carlo Ghezzi, Mehdi Jazayeri, and Dino Mandrioli ― Fundamentals of Software Engineering_, Second Edition, Pearson Education,Inc., 2003

[3] Yuan Chongzhi - Principle and Application of Petrinet [M]. Beijing: Publishing House of Electronic Industry, 2005:40-66

[4] Petri Net Model of Session Initiation Protocol and Its Verification_ - Yang Peng; Yuan Zhanting; Wang Jizeng; Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on Digital Object Identifier:10.1109/WICOM.2007.466 Publication Year: 2007; Page(s): 1861-1864.

[5] An Approach for Ensuring Security and its Verification- Suparna Karmakar, Sayani Chandra ; International Journal of Computer Science Engineering (IJCSE), ISSN : 2319-7323, Vol. 2 No.03 May 2013; pp:42-49

[6] Mayr, E.W.: An Algorithm for the General PetriNet Reachability Problem, ACM STOC, 1981, pp.238-246

[7] K. Jensen, L.M. Kristensen, L. Wells, "Coloured Petri Nets and CPN Tools for Modelling and Validation of Concurrent Systems".International Journal on Software Tools for Technology Transfer, (2007), Springer Verlag, 213-254.

[8] J.-I. Rocha, L. Gomes, and O. Dias, "Dataflow model property verification using petri net translation techniques," in Industrial Informatics (INDIN), 2011 9th IEEE International Conference on, July 2011, pp.783 –788.

[9] Well Formed PetriNet for Reachablility Yuta Ikeda; Satoshi Shimada; Takao Miura Signal Image Technology and Internet Based Systems (SITIS), 2012 Eighth International Conference on Year: 2012 Pages: 595 - 602, DOI: 10.1109/SITIS.2012.91

[10] Elhillali Kerkouche, Allaoua Chaoui, Ouassila Labbani, et al. A UML and Colored Petri Nets Integrated Modeling and Analysis Approach using Graph Transformation [J]. Journal of Object Technology. 2010.9(4):25-43.

[11] Ikeda, Y. and Miura, T.: Prototyping Color Timed Petri Nets, IEEE Pacific Rim Conference on Communications, Computers and Signal Processing(PacRim), 2011