Computational Tool for Modeling with Petri Nets

Edson R. C. Sabino
Laboratório de Dispositivos e Nanoestruturas
Universidade Federal de Pernambuco
CEP: 50670-000
E-mail: ricalado@bol.com.br

Edval J. P. Santos
Laboratório de Dispositivos e Nanoestruturas
Universidade Federal de Pernambuco
CEP: 50670-000
E-mail: edval@ee.ufpe.br

Tomaz de C. Barros
Laboratório de Dispositivos e Nanoestruturas
Universidade Federal de Pernambuco
CEP: 50670-000
E-mail: tomaz@ee.ufpe.br

ABSTRACT

Petri nets are used to model networks, automation systems, batch production systems, etc. In this paper, a Petri net player written in C is presented. It has been developed to model such systems. In the paper, our player is used to model sensors connected to a network with statistical access, such as TCP/IP. The objective of the model is to estimate the optimum number of sensors to be connected to a given node. In this work the model is presented.

Keywords

Petri net, computer tool, communication modeling protocol.

1. INTRODUCTION

With the development of faster ethernet networks, such as: fast ethernet and gigabit ethernet, this technology has again being considered for use in a industrial environment, due to its low price, second source, and well known protocols. In particular, such network technology can be used in smart sensor application for remote sensing. However, in such application it is necessary to known how many sensors can be connected to a given node, as the network is a shared resource. This kind of problem is well suited for Petri nets modeling.

As a practical example, consider fast ethernet, the data speed is 100Mbps. At this transmission speed an ICMP package can be sent in a few microseconds. Increasing the probability of finding the network free for communication. Therefore, the probability that a smart sensor don't get access to transmit data in determined period of time due to the channel be occupied decreases. This fact makes possible the use of such high-speed statistical access nets for communication among a reasonable number of smart sensors and supervisors. Besides, such protocols are open, well documented, and information about the protocol is easy to find and free. There is a need to model the communication in such networks to determine its performance as the number of sensors connected to the net increases.

To model the network performance, a simplified model of the TCP/IP protocol has been developed using Petri nets. The performance is analyzed using a computational tool also developed in our laboratory. The tool performs simulations of sensor network with different quantities of sensors.

This work is divided in four parts, first this introduction, and then the Petri net player is introduced. Third, the application of the player to model the sensor communication and finally the conclusions.

2. PETRI NET PLAYER

A Petri net is a graphic and mathematical modeling tool applicable to many systems.

A Petri net is a 5-tuple, PN=(P,T,F,W, M_0) where:

- $P=\{ p_1, p_2, \ldots, p_m\}$ is a finite set of places,
- $T=\{ t_1, t_2, \ldots, t_n\}$ is a finite set of transitions,
- $F \subseteq (P \times T) \cup (T \times P)$ is a set of arcs(flow relation),
- $W: F \Rightarrow \{1,2,3,\ldots\}$ is a weight function,
- $M_0: P \Rightarrow \{0,1,2,3,\ldots\}$ is a initial marking,

The topology of a Petri net can be described through a preconditions matrix and a posconditions matrix. The posconditions matrix is defined as being aij+ = w(i,j), where w(i,j) is the weight of the arc from transition i to its output place j. The preconditions matrix is defined as being aij- = w(j,i), where w(j,i) is the weight of the arc to transition i from its input place j

Through the matrix representation, it is possible to work these nets using a computer's tool. There are already commercial tools for this, however the source codes of these tools are not available. We decided to develop our own tool, because it can be better adapted to several kinds of applications. The program is called a Petri net Player, or Token Player.

The Player is a tool used to simulate the dynamic of a modeled system for Petri nets. The Player was implemented in C, which facilitates the modifications of the program for possible adaptations. The preconditions matrix and posconditions matrix, the initial marking and the delays associated to the transitions are recorded previously in files txt. When the program is initiated, it loads the data and displays the matrix, the initial marking, the enabled transitions with its associated delays and the options of fire of transitions. The Player still asks (not in case it is in aleatory choice mode) to choose the transition to be discharged. After the transition is fired, the program simulates the delay associated of transition, outputs the result and displays the new current marking, the enabled transition, and in that way it is followed developing the state of the Petri net. It is important to stick out that the policy of memory adopted for delays of transitions is the enabling memory. On this policy, at each transition fired, the timers of all the timed transitions that are disabled are restarted, whereas the timers of all the timed transitions that are not disabled hold their present value. The Player fires one transition for time.

3. MODELING COMMUNICATION PROTOCOL

As was described previously, the choice of the transition that will be fired is nondeterministic. This is the reason which the Petri nets are used for modeling of protocols. As the objective of that work is to model the protocols of the Industrial Ethernet, a simplified model is being developed using Petri net to model the TCP/IP protocol. Due the some peculiarities of that model, the Player has to be adapted. As an example of application of the Player in the modeling of protocols, a simplified protocol has been modeled. In this model it is only considered one receiver and one transmitter. After reception of the data, the receiver sends one acknowledgment for the transmitter. When the data is not expected, the receiver rejects it and does not send a reply (ack). If the time of reply expires, the transmitter retransmits the data. As data can be lost during communication, due to network failure. The transitions 0 and 2 correspond at transmition of the data 0 by transmitter and in a timeout, respectively. The transitions 1 and 3 are similar for data 1. The transitions 4, 5 and 6 correspond the loss of data or of some acknowledgement (ack). The transitions 7 and 8 happen when data with wrong sequence number arrives to the receiver. The transitions 9 and 10 represent the arrival of the next data package.

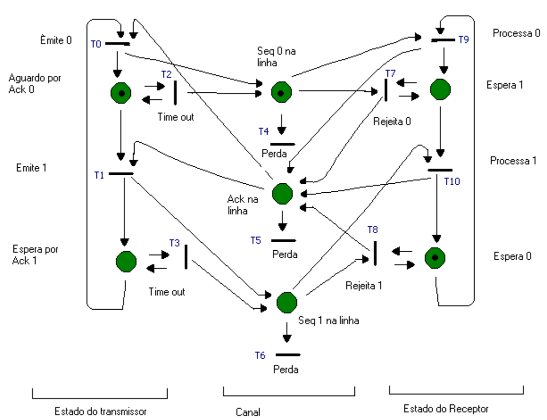

Figure 1: Petri net of the protocol described previously

Marcacao inicial
1 0 1 0 0 0 1
Transicoes Habilitadas:
t2(2) t4(0) t9(0)
Opcoes de disparo:
t4 t9
Opcao: ts=4
Continue: s / n s
Marcacao
1 0 0 0 0 0 1
Transicoes Habilitadas:

Figure 2: Part of the simulation of the previous Petri net in the Player.

4. CONCLUSIONS

The Player simulates the simplified model satisfactorily. Next, it is planned to develop a simplified model of the TCP/IP protocol over Ethernet based on Petri nets. With that, it will be possible to estimate maximum number of sensors that may be connected to a given node without causing high collision level. The Player presented can also be used for other applications, such as, the calculation of the limitation of resources in batch production systems, which is very much used in the chemical industry, beverage industry, etc.

5. ACKNOWLEDGMENTS

The authors acknowledge the support of CNPq/UFPE/PIBIC program.

6. REFERENCES

[1] Murata, T., Petri Nets: Properties, Analysis and Applications, *Proceedings of the IEEE*, 77, 541(1989).

[2] Tanenbaum, Andrew S.; Computer Networks, Upper Saddle River: Prentice Hall, 1996.

[3] Gallo, M. A. and Hancock, W.M., Comunicação entre computadores e Tecnologias de Rede, Editora Thomson, São Paulo: Pioneira Thomson Learning, 2003.

[4] Barros, T. de C., Perkusich, A., Verificação de Receitas para Sistemas deProdução em Lotes Baseada em Redes de Petri, In: XIII Congresso Brasileiro de Automática,2000, Florianópoli, SC, Brasil. Anais do XIII Congresso Brasileiro de Automática. 2000. p. 2072-2077.

[5] Maciel, P. R.M., Lins, R.D. and Cunha, P.R.F., Introdução às Redes de Petri e Aplicações, Trabalho apresentado na10ª Escola da Computação,UNICAMP, Campinas, São Paulo, Brasil, 19996.

[6] Chang, N., Kwon, W.H. and Park, J., Hardware implementation of real-time Petri-net-based controllers, Control Engineering Practice 6(1998) 889-895.