A Stochastic Petri Net Approach to Modeling and Analysis of Ad Hoc Network

Congzhe Zhang and Mengchu Zhou
Department of Electrical and Computer Engineering
New Jersey Institute of Technology
University Heights, Newark, NJ, 07102
cz3@njit.edu, zhou@njit.edu

Abstract

This paper presents a Stochastic Petri net-based approach to modeling and analysis of ad hoc wireless network. We illustrate how our model can exploit the characteristics of the system to construct a scalable model. The proposed scheme is a powerful analytical model that can be used to derive network performance much easier than a simulation-based approach. Yet it offers very close numerical results as compared with the latter. Therefore, it has great potential in assisting engineers in the design and implementation of ad hoc network.

I. INTRODUCTION

Ad hoc networks are characterized by dynamic topology due to node mobility, limited bandwidth and limited battery power of nodes. In order to analyze the performance of ad hoc networks as a function of various parameters, we present an approach for the modeling and analysis of large-scale ad hoc network systems using Petri nets. To represent network features using Petri nets, there are two requirements in advance. First, a model should be detailed enough to describe some important network characteristics that have a significant impact on performance. Second, it should be simple enough to be scalable and analyzable.

We use SPNP [1], based on Stochastic Petri Nets (SPNs), to build an approximate model for a quick numerical analysis of performance. SPNs consist of places and transitions as well as a number of functions. Enabled transitions fire according to exponential distributions, characteristic of Markov Processes. It allows the quick construction of a simplified abstract model that is numerically solved for different model parameters. We use network simulator ns2 [2] to develop a detailed simulation model to verify the accuracy and correctness of the approximate SPN model. Ns2 is a discrete event simulator targeted at networking research. It provides substantial support for simulation of TCP, routing, and multicast protocols over wired and wireless (include ad hoc network) networks. One problem when we model complex systems using ns2 is that the simulation time of these systems will increase dramatically with increasing system size. Hence, SPNs gives us an opportunity to simplify this process.

Due to advantage in quick construction and numerical analysis of Petri Nets, many related works have been done to investigate the characteristics, such as capacity and latency, of wireless network. Xiong et al. [3] modeled and simulated ad hoc routing protocol using colored Petri Nets (CPNs). They used topology approximation mechanism to solve the problem of topology changes, which is inherent character of ad hoc network. Ciardo et al. [4] modeled a scalable high speed interconnect, which is continuous hexagonal mesh like wired network, with stochastic Petri Nets. They presented both exact and tractable approximate SPN model and compared it with simulation results based on CPNs. Chen et al. [5] developed a Stochastic reward nets to analyze call admission control. They incorporated handoff call dropout time information into the decision policy, which is similar to our model.

In this paper, we illustrate how to build up an SPN model. We present this model to support our conclusion that SPNs can be successfully used for modeling scalable ad hoc networks. The rest of paper is organized as follows: Section 2 presents the structure and its approximate SPN model. Section 3 compares the numerical results of SPN model with simulation results of the ad hoc network based on ns2. Section 4 gives conclusion and discusses some features that might improve the applicability of such a model.

II. SYSTEM MODEL

An ad hoc network is a collection of wireless mobile nodes dynamically forming a local area network or other temporary network without the use of any existing network infrastructure or centralized administration. Ad hoc networks can be formed, merged together or partitioned into separate networks, without necessarily relying on a fixed infrastructure to manage the operation. In such a network, each mobile node operates not only as a host but also as a router, sending and forwarding packets to other mobile nodes in the network that may not be within direct wireless transmission range of each other. Every node in network complies with an ad hoc routing protocol that allows it to discover "multi-hop" paths, which means a packet from source node to destination node can go through several nodes throughout the network. Ad hoc routing can be classified into proactive and reactive ones based on when routes are determined. The former continuously makes routing

0-7803-7724-9/03/$17.00 © 2003 IEEE

decisions so that routes are immediately available when packets need to be transmitted with no regard to when and how frequently such routes are desired. The reactive routing determines routes on an on-demand basis: when a node has a packet to transmit, it queries the network for a route. No matter which type of routing is used, one node can begin to send packets to another node after finding a path. Due to high possibility of node mobility, a path may be redirected frequently. Thus, routing packets play as important role in network congestion as data packets do. It has a heavy effect on latency, packet drop rate, and throughput. Generally speaking, higher density node in one region means that more routing packets will be generated when we need finding a path. Even worse, one may not find the best path during routing, hereby increasing the latency time. It must be taken into account when we model ad hoc network structure.

When we model an ad hoc network, we cannot construct such structure by placing nodes into it one by one. Its size will expand too large for an exact numerical solution. We would rather describe an approximate model based on the idea of SPN decomposition. This approximate model exploits the large amount of nodes and essentially describes the behavior of one node under a workload that is generated by the whole ad hoc network. Thus the basic idea is to approximate and generate a proper amount of traffic going through one node in a network of a particular size. We use fixed-point iteration to derive results.

We will construct an approximate SPN model from incoming and outgoing subnets representing different node activities from the perspective of a single node: outgoing subnet means that packets are transferred from the current node to another node while incoming subnet means that current node are dealing with packets from outside.

Outgoing subnet is shown in Figure 1. Here we use subscript *o* and *i* to represent outgoing and incoming, respectively. Transition *Ao* generates the packets at a given rate λ and puts them into place *WBo*. An inhibitor arc with cardinality *Co* from *WBo* to *Ao* is needed to ensure that the number of packets waiting to enter the current node is finite.

Place *Buffer* contains tokens corresponding to free buffer space inside the current node. *Buffer* is shared by incoming and outgoing packets. The initial number of tokens is the total number of buffer spaces in a node. The immediate transition *GBo* reserves a buffer space for outgoing packets and put it into place *IBo*. Tokens in place *IBo* will remain in the buffer until it reaches next place *RBo*. The difference between *IBo* and *RBo* is that *RBo* receives tokens from the incoming subnet.

When a token arrives in place *RBo*, there are two possibilities at this point. Either

- Because of the shortage of buffer, physical failure during transmission, rapid moving of mobile node, or predefined timeout during waiting, the token (message) is dropped out from network and been discarded. Since we are using one node to represent the whole network, this process can happen at any time during transmission, for instance, when packets are in the source node, destination node or intermediate node's buffer. When immediate transition *DPYo* fires, the token is moved to place *RDo*, or
- Nothing happens to the token. It still remains in the buffer. When transition *DPNo* fires, the token is forwarded to place *RTo*.

The probability that a token will be dropped depends upon the size of ad hoc network and buffer, the density distribution of the nodes, the transmission rate of packets, etc. It is a variable in the SPN model. After we assign the probability to *DPYo*, we can assign the complementary probability to *DPNo*.

Timed transition *DFo* represents the completion of the dropping, after which one buffer in the current node is released by returning a token to place *Buffer*. Timed transition *TFo* means packets are transmitted successfully to another node and a token is released to *Buffer*.

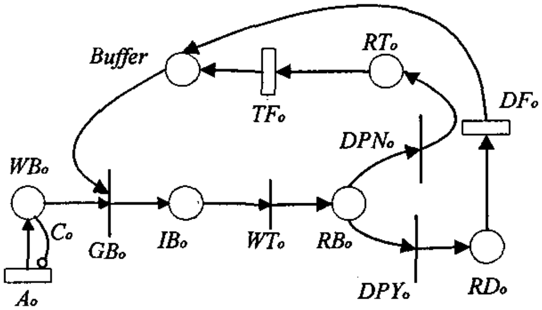

Figure 1. SPN outgoing subnet: packet transfer from the current node to another node

Incoming subnet is shown in Figure 2. Transition *Ai* generates the packets ready to be sent by neighbor nodes to current node and put the packets into place *WBi*. An inhibitor arc with cardinality *Ci* from *WBi* to *Ai* is still needed. After getting a buffer from shared place Buffer, transition *GBi* fires, and the token is transferred to place *IBi*. A token in place *IBi*, representing a packet received by the current node from its neighbors, is either destined to the current node, or has to be forwarded to other nodes further.

- If the packet has to be forwarded, immediate transition *Te* moves the token to place *RBo*, which means that incoming packet becomes outgoing packet for current node. It must be pointed out that packet generated from *Ai* also has the possibility to be dropped out throughout the transmission. Thus, incoming packet going through transition *Te* will consequently go through *DPNo* or *DPYo*.

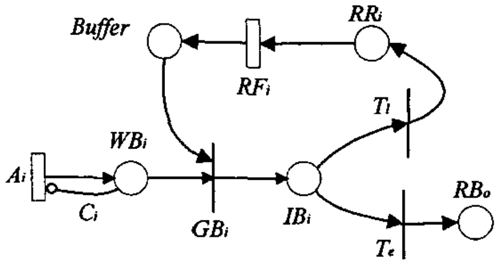

Figure 2. SPN incoming subnet: packet transfer from neighbor nodes to current node

- If the packet's destination is current node, immediate transition *Tl* moves the token to place *RRi*. Transition *RFi* represents the completion of receiving packets, after which one buffer in the current node is released by returning a token to place *Buffer*.

The probability that a packet will be forwarded or received by current node involves an approximation of transmission length. According to [6] and [7], when node density is constant, the probability density function (pdf), which means the probability of one node communicating with another node at distance x, is given by

$$p(x) = \frac{x}{\int_0^{\sqrt{A}} t\,dt} = \frac{2A}{x}$$

where A is a square network area and $\sqrt{A}$ is the maximum distance of A. Thus, the expected path length for a random traffic pattern is

$$\overline{L} = \int_0^{\sqrt{A}} x\,p(x)\,dx = \frac{2\sqrt{A}}{3}$$

Suppose that the nominal radio range for a wireless LAN is d, we get the average number of hops n required to send a packet from source to destination:

$$n = \overline{L}/d \qquad (1)$$

Hence a fraction $1/n$ of the incoming packets is directed to the current node. Note that n is a variable depending on the variation of the network area.

After getting average number of hops, the firing rate of transition Ai can be easily derived by the product of the outgoing rate λ and the average hop number n, since the packet is delivered into neighbor node for each hop it takes.

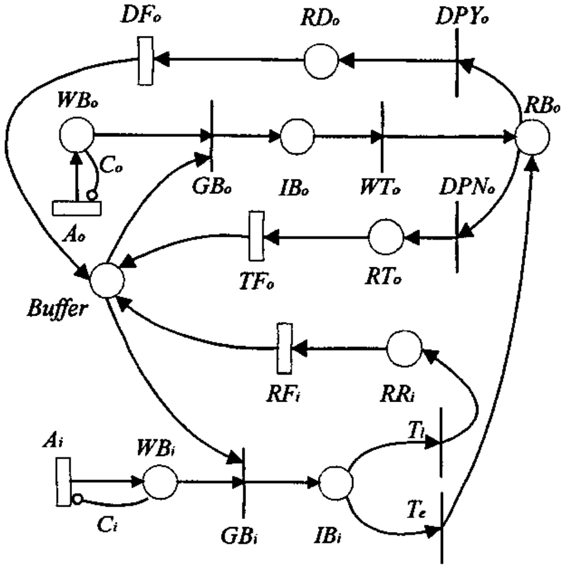

Figure 3. Overall SPN model

The composite SPN model is shown in Figure 3 that is obtained as a combination of outgoing and incoming subnet by merging shared places, *RBo* and *Buffer*. The meaning of the places and transitions in SPN is summarized in Table 1 and the firing rate and probabilities of the transitions are given in Table 2. We define *#(p)* as the number of tokens in place *p*. The priority of transitions depends on our definition. We illustrate that *GBi*'s priority is higher than *GBo*'s to ensure that the delivery of packets in transit takes priority over the injection of new packets into the network.

The undefined parameter in Table 2 is x, the average time required by an outgoing packet to obtain and fill a buffer in the destination node. This process may include several nodes on its path.

By summery, we follow the rule that only after there's an outgoing packet leaving the current node's buffer, an incoming packet can obtain a buffer slot and enter place *IBi*.

Hence, we set up the following fixed-point iteration scheme:

(1) Choose an initial guess $x(0)$ for x;

(2) Compute the successive values for x as:

$$w1*(1-\alpha)+w2*\alpha = w3 \qquad (2)$$

where w1 is the average waiting time and is obtained using Little's Law:

$$w1 = \frac{E[\text{number of packets waiting been transmitted}]}{E[\text{throughput of packets}]}$$

$$= \frac{E[\#(RTo)]}{E[rate(TFo)]}$$

Place	Meaning
Buffer	Free buffers
WBo	Outgoing packets wait for a buffer
WBi	Incoming packets wait for a buffer
IBo	Outgoing packet in the buffer
IBi	Incoming packet in the buffer
RBo	Outgoing packet that is remaining in the buffer
RTo	Ready to transmit outgoing packet
RD$_o$	Ready to drop outgoing packet
RRi	Ready to receive incoming packet

Transition	Meaning
Ao	Generate packet that is to be transmitted
Ai	Externally generate a packet that coming into the local node
GBo	Outgoing packet gets a buffer space
GBi	Incoming packet gets a buffer space
DPYo	Outgoing packet is dropped
DPNo	Outgoing packet remains in the buffer
WTo	Outgoing packet waiting in the buffer
Tl	Receive incoming packet to current node
Te	Forward incoming packet to another node
TFo	Transmitting an outgoing packet
DFo	Dropping an outgoing packet
RFi	Receiving an incoming packet

Table 1. Meaning of places and transitions in SPN model

similarly,

$$w2 = \frac{E[\#(RDo)]}{E[rate(DFo)]} \text{ and } w3 = \frac{E[\#(RRi)]}{E[rate(RFi)]}$$

when we substitute x into equation 2, $x(i+1)$ is substituted in the left side and $x(i)$ is substituted in the right side.

Transition	Firing rate
Ao	λ
Ai	$n\lambda$
TFo	$\#(RTo)/x$
DFo	$\#(RDo)\cdot x/n$
RFi	$\#(RRi)\cdot x/n$

Transition	Priority	Firing probability
GBo	4	1
GBi	5	1
DPYo	2	$1 - \alpha$
DPNo	2	α
WTo	1	1
Tl	3	$\beta = 1/n$
Te	3	$1 - \beta$

Table 2. Firing rates and probabilities of the transitions in SPN model

(3) Stop the iterations when $x(i+1)$ and $x(i)$ are sufficiently close.

III. NUMERICAL RESULTS AND COMPARISON

For our interest, we focus on the average packet delay τ, defined as the average time elapsing from the instant a packet is generated by its source node (firing of transition *Ao*), to the instant it is read by its destination node (firing of transition *RFi*). In the model of Figure 3, this is obtained as the sum of three components:

(1) the average time a packet waits before it is put into a buffer in the current node, computed using Little's law:

$$\frac{E[\#(WBo)+\#(IBo)]}{E[rate(Ao)]}$$

(2) the average time a packet waits before it is removed from the buffer in the destination node:

$$\frac{E[\#(RRi)]}{E[rate(RFi)]}$$

(3) average transmission time x, which is defined above.

We use a detailed simulation model based on the latest ns2 version. We adopt Ad hoc On-Demand Distance Vector (AODV) [8] routing protocol to deal with a routing problem. It is an on-demand protocol. There's only a minor difference if we use other routing protocols.

In our experiment, random waypoint model is used to generate node mobility model. 802.11 for wireless LANs is used as a shared-media radio with a nominal bit-rate of 2 Mb/sec and a nominal radio range of 250 meters. An ad hoc network in a field with dimensions 670m × 670m is used. Thus, from equation 1, n is 2.5 in this experiment. β can be derived from n. We cannot use a fraction in simulation because none of the packets can be divided into several parts. It is suitable when we use it to calculate β in SPNs.

In most cases, the fixed-point scheme converged in a few iterations. But if we choose an initial $x(0)$ far from normal average time, the divergence happens. Future research should explain this phenomenon and give a theoretical explanation. In ns2, in order to get an average result of one point in Figure 4, we need to run simulation at least tens of times. Comparing to the time consuming simulation, the fixed-point scheme's cost is neglectable.

There are 30 nodes roaming in this area. The speed of a node varies from 0 m/s to 20m/s to change mobility. Traffic sources are CBR, i.e., continuous bit-rate. The source-destination pairs are spread randomly throughout the network. Only 512 byte data packets are used. The number of sessions increases along with the number of nodes. All traffic sessions are established at random times and stay active until the end. Similar simulation environment has been used before in several recent performance studies on ad hoc networks [9,10].

Figure 4 shows the result of average packet delay as a function of successful delivery ratio α. Delay time becomes longer as the number of drop packets increase. We can specify the delivery ratio in SPN. We cannot predict this value in NS2. Hence the delivery ratio value in Figure 4 from SPN have a slightly difference correspond to the data from NS2.

Our experiment shows that even packet drop probability is low, which means most generated packets can arrive at their destinations successfully; system throughput is surprisingly low compared to their capacity. Related research is shown in [7].

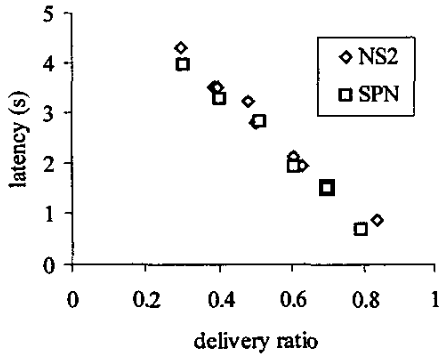

Figure 4. Successful delivery ratio vs. average latency

IV. CONCLUSION

In this article, we present a stochastic Petri net model to represent an ad hoc network. Our scheme provides a customizable approach to analyze the characteristics and performance of the system. It is shown that a close match exists between the obtained results and those from a simulation model in ns2. The proposed scheme costs negligible computational effort compared with that of a simulation method. While SPN model can give a theoretical solution for ad hoc network, ns2 is only used as a detailed model. Because ns2 has become a standard simulation tool in network research, the comparison gives us a link between Petri nets and detailed discrete event system.

Some characteristics can be obtained by SPN with slight modification of model. All of the time delays attached with transitions in our model are approximated with exponential distributions. This is not always true in a real system. For instance, sometimes delays are constants. We will apply Erlang distributions with a given mean in the SPN model to approximate the constant distribution. That will increase the computation complexity but can improve our model with better practicability.

Network security is an important issue in the current ad hoc network research. Several aspects have been stressed including routing protocols, authentication, access control, quality of service (QoS), etc. We will model our research [11] of "Security Level" concept into Petri nets to analyze corresponding ad hoc network's changes and performance.

REFERENCES

[1] G. Ciardo, K. Trivedi, and J. Muppala, SPNP: stochastic Petri net package, in *Proc. 3rd Int'l Workshop on Petri Nets and Performance Models (PNPM'89)*, pp. 142-151, Kyoto, Japan, Dec. 1989, IEEE Computer Society Press.

[2] S. McCanne and S. Floyd, NS Network Simulator. [Online]. Available: http://www.isi.edu/nsnam/ns/.

[3] C. Xiong, T. Murata, and J. Tsai, "Modeling and Simulation of Routing Protocol for Mobile Ad Hoc networks Using Colored Petri Nets," *Research and Practice in Information Technology*, vol. 12, pp.145-153, Australian Computer Society, 2002.

[4] G. Ciardo, L. Cherkasova, V. Kotov, and T. Rokicki, "Modeling a scalable high-speed interconnect with stochastic Petri nets," in *Proc. Int'l Workshop on Petri Nets and Performance Models (PNPM'95)*, pp. 83-92, Durham, NC, Oct. 1995.

[5] D. Chen, B. Soong and K. Trivedi, "Optimal call admission control policy for wireless communication networks," in *Proc. Int'l Conf. on Information, Communication and Signal Processing (ICICS)*, Singapore, Oct., 2001.

[6] J. Li, C. Blake, D. De Couto, H. I. Lee, and R. Morris, "Capacity of Ad Hoc wireless networks", in *Proc. of the 7th annual Int'l conf. on Mobile computing and networking*, pp. 61-69, Rome, Italy, July, 2001.

[7] P. Gupta and P. R. Kumar, "The Capacity of Wireless Networks", *IEEE Trans. on Information Theory*, vol. 46, issue 2, pp.388-404, March 2000.

[8] C. Perkins and E. Royer, "Ad hoc On-Demand Distance Vector Routing," in *Proc. of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, pp. 90-100, Feb. 1999.

[9] J. Broch, D. Maltz, D. Johnson, Y.C. Hu, and J. Jetcheva, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," in *the 4th annual ACM/IEEE Int'l Conf. on Mobile Computing and Networking*, pp. 85-97, 1998.

[10] C. Perkins, E.M. Royer, S.R. Das, and M.K. Marina, "Performance comparison of two on-demand routing protocols for ad hoc networks," *IEEE Personal Comm.*, vol. 8, issue 1, pp.16-28, Feb. 2001.

[11] C. Zhang and M. Zhou, "Security Enhanced Ad Hoc On-Demand Routing Protocol", *3rd Annual IEEE Information Assurance Workshop*, West Point, NY, June 2002.