

Performability Analysis of Token Ring Networks using Hierarchical Modeling

Cheul Woo Ro, Artem Park
Div. of Computer & Info. Eng.
Silla University, Busan, KOREA

Abstract

It is important for communication networks to possess the capability to overcome failures and provide survivable services. We address modeling and analysis of performability affected by both performance and availability of system components for a token ring network under failure and repair conditions. Stochastic reward nets (SRN) is an extension of stochastic Petri nets and provides compact modeling facilities for system analysis. In this paper, hierarchical SRN modeling techniques are used to overcome state largeness problem. The upper level model is used to compute availability and the lower level model captures the performance. And Normalized Throughput Loss (NTL) is obtained for the composite ring network for each node failures occurrence as a performability measure.

One of the key contributions of this paper constitutes the Petri nets modeling techniques instead of complicate numerical analysis of Markov chains and easy way of performability analysis for a token ring network under SRN reward concepts.

Keywords: Performability, SRN, hierarchical modeling, token ring network.

1. INTRODUCTION

With the increase in complexity, scale, and speed of communication networks, network performance under failure has become a great issue in industry. A failure may significantly reduce the capability of the communication network to efficiently deliver service to users. The networks need to cope with failures to preserve the network service integrity under failures. For this reason, network reliability and survivability are very importance metric for communication networks and performance metric have been studied in several papers [1-3].

Pure performance models assume failure free operation of the components involved. However failures of components within a system could lead to performance degradation. Thus, the effect of partial failures needs to be factored in to yield a comprehensive evaluation of the system. Pure availability models on the other hand only consider the failures of components. Thus, to get a comprehensive picture, a combination of performance and availability models is needed. This concept was first introduce by Meyer and called "performability"[4]. In this paper, we address a performability analysis of a token ring network. The token ring network consists of N nodes and N links connected. The failure of some nodes or link in a ring network may not necessarily bring the whole network down.

In this paper, we propose to carry out a composite approach to evaluate the network performability, where the transient overload analysis is incorporated into the failure frequency analysis. We use a hierarchical model which captures both the system transient behavior, availability to evaluate failure and repair impact on the system and the system steady state behavior, normalized throughput loss and mean response time.

2. STOCHASTIC REWARD NETS (SRN)

In order to make more compact models of complex systems, several extensions are made to Stochastic Petri Nets [5, 6], leading to the SRN [7]. One of the most important features of SRN is its ability to allow extensive marking dependency. In an SRN, each tangible marking can be assigned with one or more reward rate(s). Parameters such as the firing rate of the timed transitions, the multiplicities of input/output arcs and the reward rate in a marking can be specified as functions of the number of tokens in any place in the SRN. In a SRN model, assigning transition rates to state transitions and solving the continuous Time Markov Chain (CTMC), the probability mass function is then obtained.

Another important characteristic of SRN is the ability to express complex enabling/disabling conditions through guard functions. This can greatly simplify the graphical representations of complex systems. For an SRN, all the output measures are expressed in terms of the expected values of the

* Corresponding author: E-mail : cwro@silla.ac.kr

Manuscript received Nov. 17, 2009 ; accepted Dec. 20, 2009

reward rate functions. To get the performance and reliability/availability measures of a system, appropriate reward rates are assigned to its SRN. As SRN is automatically transformed into a Markov Reward Model (MRM) [6,7], steady state and/or transient analysis of the MRM produces the required measures of the original SRN. The measures are defined in terms of reward rates associated with the markings of the SRN.

3. HIERARCHICAL PERFORMABILITY MODELING

Integrating system availability and performance in a single model often causes the largeness and stiffness problems[8]. In order to compute performance measures for the ring network under failures and repair events, we use the hierarchical modeling approach in which the upper model describes the availability for the ring network, while the lower model characterizes the system performance under each state of the availability model [9].

Hierarchical SRN model is the Markov reward model where the reward rates comes from a sub-model. The upper level model is the structure state model representing the state of the system with regard to failure and repairs. The lower level model captures the performance of the system within a given structure state.

We build both the availability model as a upper level model and performance model as a lower level model.

3.1 Upper level model: Availability model

We use availability model as upper level model. The SRN model is shown in Figure. 1. The N tokens in places $pNup$ and $pLup$ represent the number of nodes and links correspondingly. The firing of transition tnf represents failure of a node. The '#' after transition rate λ_n means that the actual rate is dependent on the number of tokens in place $pNup$, the rate is multiplied by the number of tokens in place $pNup$. When the transition tnf fires one token moves to the place $pNdn$. Depending on the probability $1-F$ or F , which means successful bypass or not correspondingly, one of the immediate transitions $tcov$ or $tuncov$ fires. If the token moves to $pcov$ it means successful bypass and the network continues to operate. The amount of tokens in place $pcov$ represents the amount failed nodes. The firing of transition tn_rep represents node repairing. After repairing token moves to the place pnf . In case when token moves to the place $pNetdn$ the network goes down. At the moment when transition $tuncov$ fires, the token also moves from transition $tuncov$ to the place pnf .

The firing of the timed transition tlf represents the failure of a link. The firing rate λ_l is dependent on the number of tokens in place $pLup$. When transition tlf fires, one token moves from $pLup$ to $pNetdn$ and the whole network goes down. At the same moment the token also moves to the place plf .

We put inhibitor arc from $pNetdn$ to $pLup$. Without this inhibitor arc the tokens would continue to move from $pLup$ to $pNetdn$ after the network is down. The inhibitor arc from $pcov$ to $tcov$ is used to put a constraint that there must be at least one node working for the network to be up.

The token in place pnf or plf means the failure of node or link correspondingly. This is done to know the reason of network failure. The firing of transition tr_rep represents the repairing of a whole network. After firing of this transition the token moves to the place $prep$. Depending on reason of the network failure immediate transition tnr or tlr fires and token moves back to $pNup$ or $pLup$. Some arcs in this model has variable arc multiplicity, which means that if transition fires the all tokens from the source place will be transferred via corresponding transition.

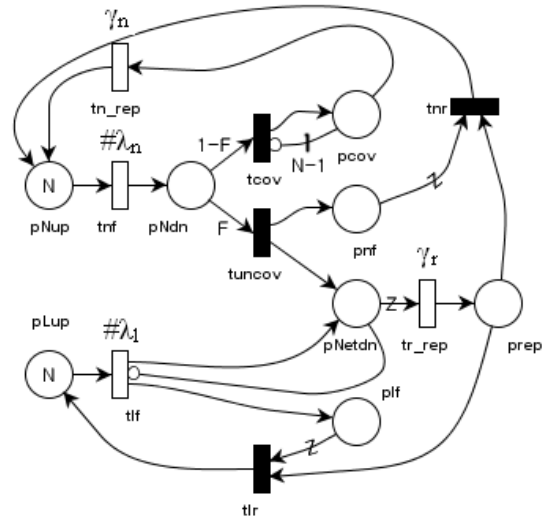


Fig. 1 SRN availability model of a ring network

3.2 Lower level model: Performance model

As a lower level model, we consider a client server system based on a token ring network consisting of N workstations (nodes) and one server interconnected by a local area network. The server station receives requests from its client stations processes the request and returns replies to the requesting stations under failure conditions.

The client server system consists of 3 subsystems which are tagged client, server, and superclient, as shown in Figure. 2. The superclient subsystem which is lump of remaining clients enables us to closely approximate $(N-1)$ multiples of single client subsystem. Using this approach we reduce the number of states of the Markov chain and thus can resolve state largeness problem [11].

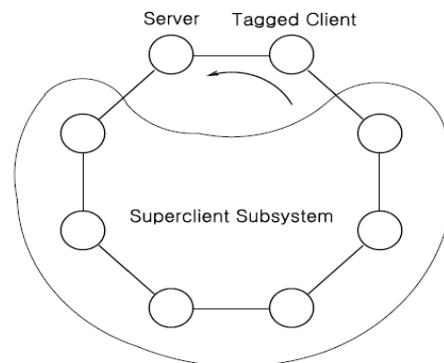


Fig. 2 token ring network subsystems

We take the SRN model from [11] as shown in Figure. 3 as

the client server system. But [11] did not include the effect of failures. In this paper, we employ this SRN model as a lower level model under failure conditions and construct hierarchical model.

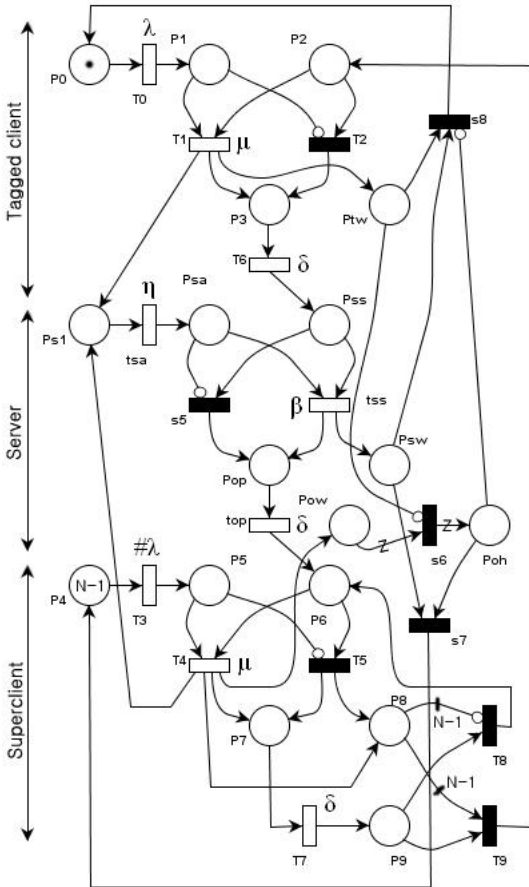


Fig. 3 SRN client-server model (lower level)

Figure. 3 shows SRN model which consists of 3 subsystems as above mentioned. The detail meanings of places and transitions in Figure. 3 are explained on Table 1 and 2, respectively.

When the number of PN tokens in P8 reaches N-1, implying that all members of the subsystem have been polled, then with a PN token in P9, T9 fires immediately and a PN token is deposited in P4.

The zigzag sign on the input arc from Pow to s6 denotes that the multiplicity of the arc is variable. Assume that Pow contains k PN tokens where 0 < k ≤ N-1, and Ptw has none. Then s6 is enabled and it fires immediately by removing all k PN tokens from Pow and k PN tokens will be deposited in Poh.

Table 1. Meanings of places in Fig. 3 SRN model

subsystem	place	meaning
tagged client	P0	the client is in idle state
	P1	client is waiting for the network token
	P2	the network token has arrived, polling token

	Ptw	the client is waiting for reply from the server
	P3	the client has finished transmitting its request
server	Ps1	the client's request has arrived to the server
	Pss	the condition that the server has received the network token and can start transmitting a reply, if one is ready.
	Psa, Pss, Pop	serve the same purpose as P1, P2, and P3 respectively
	Psw	the server has completed serving one request and a determination is to be made regarding whose request was just serviced.
	Ptw (Pow)	the tagged client (a client from the superclient subsystem) is waiting for a reply to its request.
superclient	P4	contains N-1 tokens. N is passed by higher level for each node failures occurrence
	P5, P6, P7, Pow	same role as as P1, P2, P3 and Ptw, respectively
	P8, P9	keep a count of the number of times the token has gone through P6
	Poh	denotes the number of clients from the superclient subsystem whose requests were received by the server before that of the tagged client.

Table 2. Meanings of transitions in Figure. 3 SRN model

subsystem.	transition	meaning, rate
tagged client	T0	the client has generated a request., λ
	T1	transmission time, μ
	T2	the client has no request to transmit
	T6	the network token is being passed to its neighbor, (polling time), δ
server	T4, T5, T7,	same role as as T1, T2, T6 respectively
	Tsa	the time required for the server to process a request., η
	tss	the time to transmit a reply, β
superclient	T4, T5, T7,	same role as T1, T2, T6 respectively
	s8	the tagged client's request cannot be replied until request from the superclient subsystem that arrived before it have been replied.

T_8	the network token will not leave the superclient subsystem when less than $N-1$ clients have been polled.
T_9	the tagged client now has access to the network after all members of the superclient subsystem have had a chance to transmit their requests.

4. PERFORMABILITY MEASURES

In order to obtain the performability measures numerically from the SRN model, underlying CTMC is generated and solved through the use of the well known software package, SPNP (Stochastic Petri Net Package) [12] developed by researchers at Duke university. SPNP provides support for specifying the SRN using a "C" like programming language and allows the modeler to do steady-state, transient, cumulative transient and sensitivity analysis.

We assume the all transition firing rates in our SRN models are exponentially distributed.

- Unavailability

Unavailability is quantified by the transient performance of a ring networks which are exposed to link and node failures. It can be computed using the following construction in SPNP.

```
double get_unAvail() {
    if (mark("ringfail") >= 1) return 1.0;
    return 0.0; }
```

In our code we call this function during solving the model in transient mode at each time interval.

- Normalized Throughput Loss (NTL)

Since the effect of node failures is considered, NTL is performability measure as opposed to just performance measure. The throughput of a ring network from the lower level model depends on how many stations are working under failure conditions. NTL is the fraction of throughput with i stations, the number i is passed by upper level model when i nodes are functioning and $N - i$ nodes are failed for each node failures occurrence. NTL is calculated using the following formula:

$$NTL = \frac{\sum_{i=1}^N \text{Throughput}[i] * \text{Frequency}[i] - \sum_{i=1}^N \text{Throughput}[i] * \text{Frequency}[i]}{\sum_{i=1}^N \text{Throughput}[i] * \text{Frequency}[i]}$$

where $\text{Throughput}[i]$ is the network throughput, calculated by taking the real value of t_{ss} transition's rate with available i stations, the number i is passed by upper level mode when i nodes are functioning and $N - i$ nodes are failed and $\text{Frequency}[i]$ is the number of times of calling the low-level model with particular amount of available nodes i , calculated by accumulating this number in local variable.

- Mean Response Time (MRT)

MRT is a time to take to service a job. To calculate MRT, we use the formula for probability that the tagged client is idle.

$$IdlePb = \frac{1/\lambda}{1/\lambda + \tau}, \text{ this gives us: } \tau = \frac{1 - IdlePb}{\lambda * IdlePb}$$

Where λ is a client request rate, τ is mean response time.

- throughput

The network throughput is number of reply per unit time executed by server and can be obtained as a reward rate of transition t_{ss} . It is calculated by the SPNP's build in function $rate(char * tr_name)$.

$$Throughput = rate("t_{ss}");$$

5. NUMERICAL RESULTS

5.1 Input data

We now present the input data for solving our hierarchical model. For the higher level we specify failure values as once a day ($\lambda_n=1/24$ in hour unit) and ones a week ($\lambda_l=1/168$ in hour unit), for nodes and links correspondingly. We assign repair time as 10 minutes ($\gamma_n=6$ in hour unit) and 1 hour ($\gamma_r=1$ in hour unit) for nodes and the whole ring network correspondingly. And finally we put probability 0.3 that the node fails without successful bypass which means that it cause ring failure and the probability 0.7 for the successful bypass. For the lower level model, we assume that network bandwidth is 100 Mbps and network cable length is 2 Km and that request packet length is 1500 byte which corresponds to $1/\mu = 0.12$ ms, $\mu = 8.3$, and that reply packet length is 6K bytes/packet which corresponds to $1/\beta = 0.48$ ms, $\beta = 2.08$, and that the mean token length is 30 bytes, which corresponds to a mean polling time $1/\delta = 1.0/(0.01/(N+1) + 0.0024)$ ms, where N is a number of nodes. Finally we assume that the mean time taken by the server to produce a reply is 2 ms ($\tau=2$).

5.2 Numerical results

The results of unavailability analysis obtained by solving higher level SRN model are shown in Figure. 4.

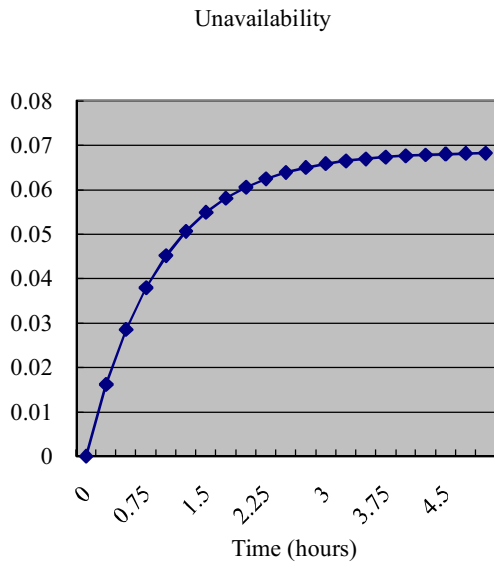


Fig. 4. Unavailability of the ring network

We tabulate the results of NTL for different number of nodes and different token requesting rates to see the changes of NTL values as shown in Table 3..

Table 3. Normalized Throughput Loss

N	Normalized Throughput Loss (NTL)			
	$\lambda = 1$	$\lambda = 2$	$\lambda = 3$	$\lambda = 4$
3	0.1989915	0.157456	0.141119	0.132438
4	0.183924	0.143073	0.127887	0.119952
5	0.162672	0.125141	0.111541	0.104485
6	0.142798	0.109158	0.097127	0.090909

The results of throughput, number of reply per millisecond, are shown in Figure. 5. As λ and N increasing, network throughput is increasing, however if the number of

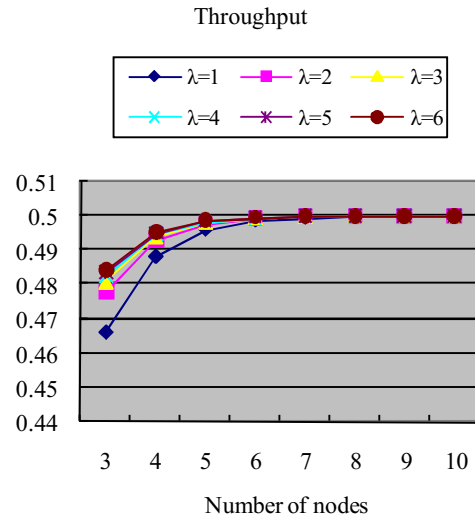


Fig. 5. Network throughput

nodes is greater than 7 the throughput is not increasing. It stays the same even we increase requesting rate.

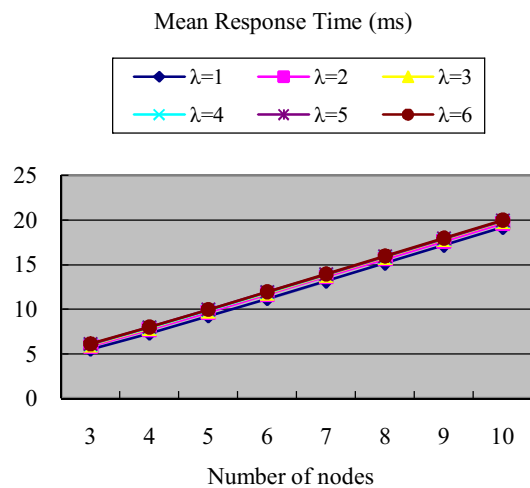


Fig. 6. Mean Response Time (ms) of a tagged client

Figure. 6 shows the results of mean response time of a tagged client in milliseconds. As we see the MRT depends mostly on the number of nodes.

5.3 How do we run the hierarchical model.

To build the hierarchical model we create two separate independent models written in special “C” like programming language, which is used to describe SRN models ready to analyze with SPNP package. The one file represents higher level and the other one low level model. The higher level file contains the code which calls low level model at some step of its execution. For the calling the low level model we use C-shell script, which we prepare by setting some parameters for the low level model and then call it using system function. The example of this call is shown in Listing 1. After the low level model called it is then executed and we need to get the results

to the higher level. This is done by using pipes and language for processing text-based data, for example *awk*. The output of the low level model is written to standard output file with extension *.out. We use this file to read the returned values. In higher level model we open the pipe and using *awk* read the necessary data and then we put it to the local variable. We define guard function to call low level model for the transition **tnf** and firing of this transition causes the changes of the number of nodes which is our interest. The guard function is called when **tnf** is enabled.

Listing 1. Calling low level model.

```

...
fshell = fopen("l_shell", "w");
fprintf(fshell, "set n = %d\n", nnodes);
fprintf(fshell, "set rfail = %d\n", ring_fail);
fprintf(fshell, "set lambda = %d\n", lambda);
fprintf(fshell, "echo $n $rfail $lambda | 16.spn");
fprintf(fshell, "\n");
fclose(fshell);
system("csh l_shell");

if ((pp = popen("grep TSS_THP 16.out | awk '{print $4}' ", "r")) ==
NULL) {
    perror("popen");
    exit(1);
}
fscanf(pp, "%14lf", &val);
fclose(pp);
printf("Value from lower level:%f\n\n", val);
if(ring_fail ==0) Thp[nnodes] = val;
    
```

6. CONCLUSION

We constructed hierarchical performability SRN models for a token ring network and analyzed the network performability involves both the system availability and the system performance under failure conditions. Measures of interest, such as NTL as a performability measure, unavailability from the higher level, MRT and throughput from the lower level were obtained by using reward rates in SRN models.

REFERENCES

[1] D.Y.Chen, S.Garg, and K.S. Trivedi, "Network survivability performance evaluation: A quantitative approach with applications in wireless ad-hoc networks," ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM'02), Sep. 2002. ACM.

[2] P.Heegaard, K.S.Trivedi, "Survivability Quantification of Communication Services," *IEEE DSN 2008, Alaska*, 2008.6.

[3] Yun Liu, K.S. Trivedi,"Survivability Analysis of Telephone Access Network," *International Journal of Performability Engineering, Vol.2, No.1*, pp 29-44, 2006. 1.

[4] J.F. Meyer, "On evaluating the performability of degradable computing systems", *IEEE Trans. Computers, vol C-29*, 1980 Aug, pp 720-731.

[5] M. Molly, "Performance analysis using stochastic Petri nets," *IEEE Trans. Computer, Vol.C-31*. 1982 Sep. pp 913-917.

[6] Marsan, M.A. Conte, Balbo, "A class of generalized stochastic petri nets for the performance evaluation of multiprocessor systems," *ACM Trans. Computer Systems*, 2:93-122, 1984.

[7] Malhotra, M. and Ciardo, G., "Dependability Modeling Using Petri-Net," *IEEE Transactions on Reliability, Vol.44, No.3*, pp.428-440, Sept., 1995.

[8] A. Bobbio, K.S. Trivedi, "An aggregation technique for the transient analysis of stiff Markov chains," *IEEE Transactions on Computers* 35 (9), 1986, pp. 803–814.

[9] R.M. Smith, A. Ramesh, K.S. Trivedi, "Performability analysis: Measures, an algorithm and a case study," *IEEE Transactions on Computers* 37 (4), 1988, pp. 406–417.

[10] G. Ciardo and K. S. Trivedi, "A Decomposition Approach for Stochastic Petri Net Models," *International Conference on Petri Nets and Performance Models, Melbourne, Australia*, December 1991, pp. 74-83.

[11] Oliver C.Ibe, K.S. Trivedi, "Performance Evaluation of Client-Server Systems," *IEEE Transactions on Parallel and Distributed Systems*, 4 (11), 1993, pp. 1217-1229

[12] Ciardo, K.S. Trivedi, *SPNP Users Manual*, Duke University, 1994.



Cheul Woo Ro

He received the PhD degrees in Computer Engineering from Sogang Univ, Korea in 1995. He was a member of ETRI for ten years. His main research work was development of

TDX exchange system. He is currently a professor in the department of computer engineering at Silla university. His main

research interests include modeling and analysis of communication systems.

Artem Park

He received BS degree at Uzbekistan. He was a member of UNDP programmer. He is currently a graduate student at Silla university. His main interests include modeling and reliability analysis of computer system and pattern recognition.