2012 5th International Conference on BioMedical Engineering and Informatics (BMEI 2012)

Estimating Network Path Loss Episode Frequency by Passive Measurement

Hua Wu
School of Computer Science & Engineering Southeast University
Nanjing, China
Email: hwu@njnet.edu.cn

Jian Gong
School of Computer Science & Engineering Southeast University
Nanjing, China
Email: jgong@njnet.edu.cn

Ziyu Ma
Nanjing Foreign Language School
Email: Judy_922@126.com

***Abstract*—End-to-end loss episode frequency is the congestion characteristic of the network path, but it is difficult to measure passively. Although there are active methods to measure this metric, probe packets may have impact on the network in order to get accurate result. In this paper, a passive method is proposed to estimate the packet loss episode frequency of end-to-end network path by parallel TCP flow trace. We evaluate the accuracy of the method by simulations. The results show that the algorithm can estimate the loss episode frequency of end-to-end path within tolerable error range.**

***Keywords*—Network performance;Loss episode;Passive measurement**

I. Introduction

Packet loss characteristic is a critical element for network performance evaluation. It has significant impact on the performance seen by the users. Most of the TCP congestion control algorithms take packet loss as the Network congestion signal and adjust the sending rate according to it.

The most commonly used metric of packet loss is packet loss rate, it is the number of lost packets divided by the total number of arriving packets over a given time of duration. But the bursty characteristic [1] [2] of packet loss also has significant importance to the performance evaluation. The loss pattern is one of the key elements that determine the performance observed by the users for certain real-time applications such as packet voice and video application. For the same loss rate, two different loss patterns could potentially produce widely different perceptions of performance. RFC–3357 has defined a loss burst as the sequence of consecutive lost packets. This definition focus on the pattern of the packet loss distribution, but the background of the network congestion is ignored. Thus the relationship between the packet loss pattern and path congestion state cannot be determined.

In [3], packet loss characteristic was given a "router-centric" view. For a network path, it could be considered as a logic link with only one bottleneck at router R for a single output link as Fig. 1. A set of flows that pass through R will compete for the output link. If the arrival rate exceed the output bandwidth and the queue of R is full, the arriving packets will be dropped. Usually it means congestion occurred. The congestion state will last for some time, until the packets arrival rate drops to a certain extent. Such a process is a network path loss episode.

Path loss episode is the reflection of the path congestion state. But the measurement of this phenomenon is not as easy as the packet loss rate. For a LAN administrator, it is impossible to monitor the queue states of the backbone routers directly.

In this paper a method which can estimate the path loss episode frequency via passive measurement is proposed. The estimator is made by the transfer characteristics of two parallel TCP flows. So it's possible for the LAN administrator to provide the fine-grained performance management oriented to network path. Compared to other passive methods that measure path loss episode frequency, the biggest advantage of this method is there is no requirement for the deployment of distributed monitors, thus make it a practical method. NS2 experiments are carried out to validate the algorithm. The results showed that the algorithm can give the path loss episode frequency within tolerable error range.

This paper is organized as follows. In Section II, we present the work related to our proposal. In Section III, we provide the definitions of path loss episode, flow loss episode and the conjoint loss episode, the relationships between the path loss episode and the flow loss episode is analyzed. In Section IV, three assumptions are proposed, then the algorithm is presented and verified with NS2. In Section V, the conclusions and future directions for this work are discussed.

II. Related Work

Packet loss characteristics are among the most important network behaviors. The definition and measurement of the packet loss metrics have been the subject of study for a long time. A series of packet loss metrics are defined in RFC-2860, RFC-2861 and RFC-3357, and the corresponding active measurement methodologies are advised. ZING [5] is a tool for measuring one-way packet loss according to the advice in RFC-2860. ZING sends UDP packets at Poisson-modulated intervals with fixed mean rate. Paxson [6] recommended the use of Poisson-modulated active probes to reduce bias in the measurements, it has been a foundation for the active measurement of end-to-end delay and loss. But the accuracy of active methods has been doubted, atuhors in [3] point out that simple Poisson probing is relatively ineffective at measuring loss episode frequency. Since there are many flows share the bandwidth at the bottleneck, an active measurement is a sampling of all the dropped packets, the result is effected not only by the end-to-end congestion state, but also by the sampling method, the transfer protocols, and the characteristics of the measurement points. Furthermore, packet loss is a rare event on the path, to gain the accurate estimates need the measurement be taken a long time or send the probes at high rate, thus will skew the results.

In [7], the active and passive methods for measuring packet loss are compared, experiment results showed that the two kind of methods have uncorrelated results, this indicates that active method can hardly give the same packet loss characteristic experienced by the end users. Schormans [10] did some studies on the problem of measuring IP QoS by active method, experiment results indicate that in the real environment, in order to get the same experience with the end users, the sending rate, packet size and sending pattern of the probes packets should be considered very carefully, and the measurement equipment should be configured carefully. Otherwise, the measurement accuracy will be very poor. Comparatively speaking, passive method can give more similar result as what the end users experienced.

To measure packet loss in the passive way, the simplest method is to subtract the number of packets arrived at the destination from the number of packets that were originally sent. Distributed monitors are required for this method. Although there have been a number of passive measurement studies on the different distributed measurement infrastructures [4][8], the acquisition and storage of packet level data is high-budget to the high speed network. This kind of methods can be used in the network management systems widely because of the high deployment and management costs. In [9], the sampled flow level statistics collected by the routers are used to estimate the one-way loss. Even if the error caused by sampling can be ignored, the data acquisition is impossible to a network administrator, people can hardly get the operation data from routers outside the management domain of their own.

In [3], Joel Sommers has proposed an active method to estimate the end-to-end loss episode frequency. The main difference between his work and our proposal is that we estimate it by passive method with data collected from a single point. Without any influence on the network performance, and with the low implementation cost, our method provides a feasible proposal to estimate the loss episode frequency with low-budget.

III. Packet Loss Episode Definitions

A. Definitions

Similar to the simple model in [3], we consider the network path as a logical link with only one bottleneck router R. During the congestion period, sometimes the router queue may have free space for the arriving packets, thus some packets can be successfully transfered. This phenomenon increases the difficult in the identification of the loss episodes. In [4], the classic packet loss burst definition was enhanced based on the packet loss density in the router. Our concept is similar to that, but we use the queue occupancy as the indication of the congestion state. In order to identify the path loss episode from the "router-view", we define the path loss episode according as follows:

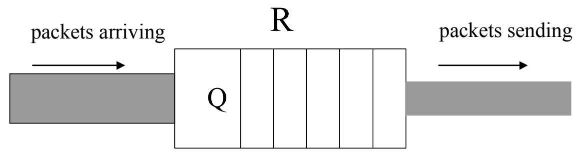

Fig. 1. A logical link with only one bottleneck router.

Definition 1: A path loss episode is identified by the following rules:

(1) A path loss episode begins when the previous loss episode had ended, and a packet loss happens at bottleneck router R;

(2) A path loss episode will last if the queue occupancy is above the threshold α, and the time since the last lost packet is not larger than β;

(3) A path loss episode ends when the queue occupancy is below α or time β has passed since the last dropped packets.

In this paper, α is set to be 80% of the queue occupancy, the setting of β need to take account of the real environment. For most of the common TCP congestion control algorithms, the sender reduce its sending rate when it senses a packet loss. This is one of the main reasons that the path loss episode will end. According to this, from the time that a packet is lost at the bottleneck, the sender senses the packet loss later and reduces the sending rate, to the time the reduced sending rate has an effect on the bottleneck, the time duration is about one RTT. In this paper, we set β to be the weighted average RTT of all the packets pass through.Following the identification rules above, the path loss episode can be identified.

The purpose of this paper is to estimate the path loss episode through the packet loss characteristics of flows, we give two related definitions of IP flows subsequently.

Definition 2: A flow loss episode is identified if among the dropped packets of a path loss episode, one or more packets belong to a specific flow.

Definition 3: A conjoint loss episode is identified if there are two or more flow loss episodes happen in the same path loss episode.

B. Relationship Between Path Loss Episode and Flow Loss Episode

The relationship between flow loss episode and path loss episode is analyzed in this section. Fig. 2 gives the relationship between them.

In Fig. 2, Q is the set of all the packets which arrive in one path loss episode. N is the set of the packets arrive in this path loss episode which belong to a flow (in this paper, we use TCP flows to estimate, we call this flow TCP_i), M is the set of all the packets which are lost during this path loss episode. For all the packets in N, $O = N \bigcap M$ is the set of the lost

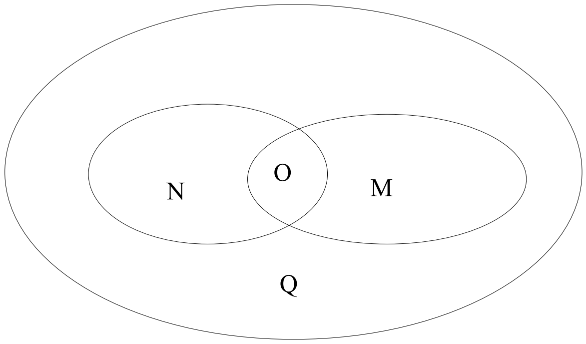

Fig. 2. Schematic diagram of the relationship between path loss episode and flow loss episode.

packets which belong to TCP_i in this path loss episode. Let $q = Number(Q)$, $n = Number(N)$, $m = Number(M)$ and $o = Number(O)$, q stands for the number of all the arrived packets in this path loss episode, n stands for the number of the arrived packets belong to TCP_i in this episode, and m stands for the number of all the dropped packets in this path episode. Being a path loss episode, $m > 0$ is a necessary condition. There are the following three possible relationships between o and m:

(1) $0 < o < m$: There is a flow loss episode of TCP_i in this path loss episode, and there are also other flow loss episodes in this path loss episode. It is a conjoint loss episode.

(2) $0 < o = m$: There is only one flow loss episode of TCP_i, and all the lost packets belong to TCP_i.

(3) $o = 0$: There isn't a flow loss episode of TCP_i in this path loss episode, TCP_i has no lost packet in this path loss episode.

As shown in Fig. 2, $\frac{n}{q}$ is the probability that one of the packet in Q belongs to N, $(1 - \frac{n}{q})$ is the probability that one of the packet in Q does not belong to N, $(1 - \frac{n}{q})^m$ is the probability that m packets in Q do not belong to N, it is also the probability that all the m lost packets do not belong to N. Therefore $1 - (1 - \frac{n}{q})^m$is the probability that m packets in Q have one or more packets belong to N. $1 - (1 - \frac{n}{q})^m$ is also the probability that TCP_i will encountered the path loss episode:

$$p = 1 - (1 - \frac{n}{q})^m \tag{1}$$

Whether a flow loss episode happened in a certain path loss episode is a probability event. It depends on the total number of the arriving packets, the number of the lost packets, and the number of the arriving packets belong to this flow.

If a packet of a flow is lost, it must be happened in a path loss episode, but not all the packets arrived at the path loss episode will be lost. We use two parallel TCP flows to estimate the path loss episode in this paper.

IV. Estimating Path Loss Episode Frequency through Parallel TCP Flows

In this paper, loss episode frequency is estimated by passive measurement. In order to simplify the derivation of the model, three assumptions are established.

A. Basic Assumptions

Assumption 1 : A packet belongs to a certain TCP flow or not and this packet will be dropped or not are independent events.

Assumption 2 : A TCP flow loss episode has the same probability to occur in all the path loss episode.

Assumption 3 : The probability of every arriving packet to be dropped in each path loss episode is the same.

Assumption 1 assumes that a packet belongs to certain TCP flow or not and this packet will be dropped or not are independent events. But this assumptions is not valid in some cases. For example, if there are TCP BIC [11] flows and Reno flows in the same environment, since BIC is a TCP congestion control algorithm designed for high-speed network , in the same network environment, BIC flows normally have larger CWND than Reno flows. Suppose there is a BIC flow and a Reno flow between two access points at the same time. We call a BIC flow which has larger CWND as TCP_l, a Reno flow which has smaller CWND as TCP_s. During one RTT time, TCP_l will send more packets than TCP_s. If the router drops every packet with the same probability, the lost packet has a larger probability belongs to TCP_l than belongs to TCP_s.

In order to satisfy assumption 1, the selected data should belong to the same end points (to ensure the packets have the same congestion control algorithm), and the transfer process happened at the same time (to ensure the packets have the most closely network environment). It happens when multi-thread downloading occurs. In this case, these flows have the same network environment, the same operating platform and the same protocol. Only the port on the client is different. By selecting the data according to these criteria, TCP flows which support Assumption 1 can be selected. We call these flows are parallel flows in this paper.

Definition 4: Flows are called parallel flows if the following conditions are met:

(1) The source and destination are the same;

(2) The flows transmitted simultaneously;

(3) The transfer paths are the same.

Assumption 2 assumes that the flow loss episode has the same probability to occur in all the path loss episodes. During a TCP flow transmission, there may be a number of path loss episodes. Because of the different background traffic, the characteristics of these path loss episodes are different, including the time duration, number of packets dropped, this lead to the different probabilities of the flow loss episodes to occur in every path loss episode. The path loss episode which have longer duration and more dropped packets will has higher probability to be encountered. In order to satisfy Assumption 2, the characteristics of each loss episode should be the same. In the real environment, this is uncontrollable. But for a small time period, the path condition is relatively stable,the error caused by this assumption can be ignored.

Assumption 3 assumes that the probability of every arriving packet to be dropped in each path loss episode is the same. In fact, the packets arrive at the bottleneck in sequence. Suppose there are two TCP flows which we call TCP_i and TCP_j. If TCP_i have packet lost before TCP_j, and the two flows have approximate the same number of arriving packets, that is $n_i \approx n_j$. $1-(1-\frac{n_i}{q})^m$ is the probability that TCP_i has a packet lost during the loss episode, after one packet has been dropped, the probability changed to $1-(1-\frac{n_j}{q})^{m-1}$. If m is large, the difference can be ignored, if m is small, especially when m equals to 1, the difference can not be ignored.

In summary, Assumption 1 can be satisfied by selecting suitable data (parallel flows), the other two assumptions may cause errors related to the network environment during the transmission, the errors caused by the two assumptions can be ignored in some conditions.

B. Estimation Algorithm

With two parallel TCP flows, let N_i denote the number of flow loss episodes of TCP_i, N_j denote the number of flow loss episodes of TCP_j, $N_{i\&j}$ denote the number of conjoint loss episodes which TCP_i and TCP_j all have packet lost. N_{path} denote the number of the path loss episodes during the parallel TCP flows transmission. N_i. N_j and $N_{i\&j}$ are the parameters that can be obtained from the TCP trace. N_{path} can not be obtained directly, it is what we want to estimate through N_i, N_j and $N_{i\&j}$.

For the parallel TCP flows TCP_i and TCP_j, it is reasonable to assume TCP_i and TCP_j have the similar number of packets arrived at the bottleneck during the path loss episodes, we suppose they all equal to n. Based on (1), the probability that TCP_i and TCP_j encountered the same path loss episode is $p = 1-(1-\frac{n}{q})^m$. The number of flow loss episodes of these flows can be expressed as :

$$N_i = N_j = N_{path}(1-(1-\frac{n}{q})^m) \tag{2}$$

According to (1), the probability that TCP_i or TCP_j will lose packet in the path loss episode is $p = 1-(1-\frac{n}{q})^m$, the probability that two flows both will lose packets in the path loss episode is $p^2 = (1-(1-\frac{n}{q})^m)^2$. During the transmission, the number of the conjoint loss episodes that TCP_i and TCP_j both have packets lost is:

$$N_{i\&j} = N_{path}(1-(1-\frac{n}{q})^m)^2 \tag{3}$$

Substitute (2) into (3), we have:

$$\begin{aligned} N_{i\&j} &= N_{path}(1-(1-\frac{n}{q})^m)(1-(1-\frac{n}{q})^m) \\ &= \frac{N_i N_j}{N_{path}} \end{aligned} \tag{4}$$

Thus the estimator of N_{path} is :

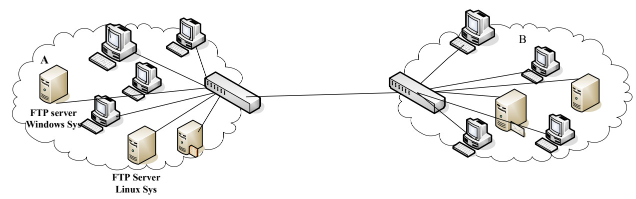

Fig. 3. Simulation environment topology.

TABLE I
TRAFFIC RATIOS OF THE THREE SCENARIOS

	Ratio of the Parallel Flows		Ratio of the background	Ratio of the background
	BIC	Reno	TCP traffic	UDP traffic
Scenario1	9.5%	5.2%	41.8%	43.5%
Scenario2	8.8%	4.8%	41.6%	44.8%
Scenario3	6.3%	3.5%	28.2%	62.0%

$$\hat{N}_{path} = \frac{N_i N_j}{N_{i\&j}} \tag{5}$$

If the number of conjoint loss episodes is 0, it implies TCP_i and TCP_j have lost packets in different path loss episode. that is $N_{i\&j} = 0$ in (5), the equation dose not hold in this case, so it is a precondition that $N_{i\&j} \neq 0$. If the selected parallel flows can not satisfy this condition, the flow trace can not be used to make the estimation.

In order to make the estimation results comparable, we can convert the estimation of N_{path} to the loss episode frequency in standard unit of time. Suppose the duration of the two parallel TCP flows transmission is T seconds, the path episode frequency estimation in one second is :

$$\hat{F}_{path} = \frac{N_i N_j}{N_{i\&j} T} \tag{6}$$

C. Simulation Environment

We validate this algorithm via the NS2 network simulator. Spindle-shaped topology is used in the simulation. As shown in Fig. 3, there are two FTP application senders at the left side, and two receivers at the right side.The congestion control algorithms used by the two FTP applications are Reno and BIC respectively, corresponding to the Windows platform and Linux platform.

In addition to the two FTP applications, the background TCP and UDP traffic are set to simulate the real scenario. Since there are no common conclusions about the traffic pattern, we use three UDP generators, they are Exponential, Pareto and CBR(constants Bit Rate with random noise). The traffic ratios of the three scenarios are show in TABLE I.

The ratios of the background traffic are different. The diversity of the background traffic can verify the general applicability of the algorithm.

The analyzed TCP parallel flows use the same congestion control algorithm, Reno and BIC are used respectively. The

TABLE II
VALIDATION USING TCP RENO FLOWS

	$\mathbf{N_i}$	$\mathbf{N_j}$	$\mathbf{N_{i\&j}}$	$\mathbf{\hat{N_{path}}}$	$\mathbf{N_{path}}$	**err**
Scenario1	95	101	60	160	174	-8.0%
Scenario2	96	100	60	160	171	-6.4%
Scenario3	92	98	43	210	205	2.4%

TABLE III
VALUDATION USING TCP BIC FLOWS

	$\mathbf{N_i}$	$\mathbf{N_j}$	$\mathbf{N_{i\&j}}$	$\mathbf{\hat{N_{path}}}$	$\mathbf{N_{path}}$	**err**
Scenario1	116	118	85	161	174	-7.5%
Scenario2	116	128	96	155	171	-9.4%
Scenario3	133	129	86	199	205	-2.9%

behavior patterns of these two congestion control algorithms have large difference. We use different congestion control algorithms to evaluate if the accuracy is affected by the congestion control algorithm.

D. Validation of the Basic Algorithm

At the three scenarios, two Reno parallel TCP flows and two BIC parallel flows are selected according to the select criterion mentioned in IV-A. The relative error is computed by the following formula:

$$err = \frac{\hat{N_{path}} - N_{path}}{N_{path}} \times 100\% \tag{7}$$

The transmission time in the NS2 experiments are 500s. When this method is used, the required monitor time is different in different network environment. The major determining factor is there must have a number of conjoint loss episodes of the parallel flows. We assume the number of conjoint loss episodes at least larger than 10 to reduce the error caused by sampling error in practice.

It is shown in TABLE II and TABLE III, our method can estimate the path loss episode frequency within tolerable error range. The difference of the TCP congestion control algorithm have no obvious effect on the accuracy of the results.

V. CONCLUSIONS AND DISCUSSIONS

We have presented a passive method to estimate the packet loss episode frequency of end-to-end network path with the trace of parallel TCP flows. The simulation experiments show that the algorithm can estimate the loss episode frequency of end-to-end path within tolerable error range.

Although there are other active and passive methods which had been proposed to estimate the path loss episode frequency, they are not as practical as our method. The implementation of our method have no impact on the operation of network, and do not need the deployment of distributed monitors. With this method, the network administrator can get the congestion frequency oriented to end-to-end path, rather than the traditional ports management based on SNMP.

However, there are a number of issues remain. Firstly,the algorithm is proposed on three assumptions. Since the real network environment is not fully consistent with the assumptions, estimation errors caused by these assumptions should be analyzed in more detail. Secondly, although our method has been evaluated in the simulation environment, the lack of experiments in the real network makes this study not fully completed. Unfortunately, experiments in the real backbone is difficult to carry out, the variation of all the queues along a path need to be monitored to validate this method. We are considering to set up a laboratory environment to evaluate our method in a controlled environment. Finally, more details need to be considered when applying our method, such as setting the parameters according to the network environment dynamically, more accurate identification of the flow loss episodes.

ACKNOWLEDGMENT

This work was sponsored by he National Grand Fundamental Research 973 program of China under Grant No.2009CB320505, the National of Nature Science Foundation of China under Grant No. 60973123, the Technology Support Program (Industry) of Jiangsu Province under Grant No.BE2011173, Jiangsu Province Key Laboratory of Network and Information Security under Grants No.BM2003201 and Key Laboratory of Computer Network and Information Integration (Southeast University) of Ministry of Education.

REFERENCES

[1] Y. Zhang, N. Duffield, V. Paxon, and S. Shenker, "On the constancy of Internet path properties," Proc. ACM SIGCOMM Internet Measurement Workshop, San Francisco, CA, USA, November 2001.

[2] M. Borella, D. Swider, S. Uludag, and G. Brewster, "Internet packet loss: Measurement and implications for end-to-end Qos," In International Conference on Parallel Processing, August 1998.

[3] J. Sommers, P. Barford, N. Duffield, and A. Ron, "Improving Accuracy in End-to-end Packet Loss Measurement," Proc. of ACM SIGCOMM, 2005.

[4] René Serral–Gracià, Albert Cabellos–Aparicio, and Jordi Domingo–Pascual, "Packet loss estimation using distributed adaptive sampling," In End-to-End Monitoring, Apr 2008.

[5] J. Mahdavi, V. Paxson, A. Adams, and M. Mathis, "Creating a Scalable Architecture for Internet Measurement," Proc. INET'98, Geneva, July 1998.

[6] V. Paxson, "End-to-End Internet Packet Dynamics," Proc. SIGCOMM 1997, Cannes, France, 139–152, September 1997.

[7] P. Barford and J. Sommers., "Comparing probe– and router–based packet loss measurements," IEEE Internet Computing, September/October 2004.

[8] D. Papagiannaki, R. Cruz, and C. Diot, "Network performance monitoring at small time scales," Proc. of ACM SIGCOMM, 2003.

[9] Yu Gu, Lee Breslau, Nick Duffield, Subhabrata Sen, "On Passive One-Way Loss Measurements Using Sampled Flow Statistics," Proc. of INFOCOM 2009.

[10] J. Schormans and J. Pitts, "So you think you can measure IP QoS?," Telecom–munications Quality of Services: The Business of Success, 2004.

[11] Xu Lisong, Harfoush Khaled, Rhee Injong, "Binary increase congestion control (BIC) for fast long–distance networks," Proc. of INFOCOM 2004.