IEICE TRANS. INF. & SYST., VOL.E93–D, NO.5 MAY 2010

PAPER

Identifying High-Rate Flows Based on Sequential Sampling

Yu ZHANG[†,††a)], **Binxing FANG**[††,†b)], *Nonmembers*, *and* **Hao LUO**[††c)], *Member*

SUMMARY We consider the problem of fast identification of high-rate flows in backbone links with possibly millions of flows. Accurate identification of high-rate flows is important for active queue management, traffic measurement and network security such as detection of distributed denial of service attacks. It is difficult to directly identify high-rate flows in backbone links because tracking the possible millions of flows needs correspondingly large high speed memories. To reduce the measurement overhead, the deterministic 1-out-of-k sampling technique is adopted which is also implemented in Cisco routers (NetFlow). Ideally, a high-rate flow identification method should have short identification time, low memory cost and processing cost. Most importantly, it should be able to specify the identification accuracy. We develop two such methods. The first method is based on fixed sample size test (FSST) which is able to identify high-rate flows with user-specified identification accuracy. However, since FSST has to record every sampled flow during the measurement period, it is not memory efficient. Therefore the second novel method based on truncated sequential probability ratio test (TSPRT) is proposed. Through sequential sampling, TSPRT is able to remove the low-rate flows and identify the high-rate flows at the early stage which can reduce the memory cost and identification time respectively. According to the way to determine the parameters in TSPRT, two versions of TSPRT are proposed: TSPRT-M which is suitable when low memory cost is preferred and TSPRT-T which is suitable when short identification time is preferred. The experimental results show that TSPRT requires less memory and identification time in identifying high-rate flows while satisfying the accuracy requirement as compared to previously proposed methods.

key words: *traffic monitoring, high-rate flow, identification, sequential sampling*

1. Introduction

In this paper, we address the problem of fast identification of high-rate flows in very high speed backbone links. Identifying high-rate flows is an important aspect of active queue management, traffic measurement and network security. As all we know, current Internet has no mechanism for controlling the throughout of each flow, which is performed by end hosts using TCP. As a result, the packet-sending rates of UDP flows or malicious TCP flows will not be reduced even when packet dropping is detected. In order to provide fairness in networks, active queue management is proposed. The main idea is to identify high-rate flows and selectively drop their packets during times of congestion. The traffic study has shown that even though there are a large number of flows in the network, a significant fraction of the traffic is carried by a small number of flows. The flows with rates in the highest 10% can constitute as much as 30%–90% of all traffic transmitted [1]. Therefore, only dropping the packets of a small number of high-rate flows will effectively improve the network traffic congestion. Furthermore, by defining a flow by the destination IP address and the destination port number, a sudden increase in a flow can be a sign of distributed denial of service (DDoS) attacks.

A naive method to identify high-rate flows is to keep per-flow counter for each arriving flow and identify flows of which the counter is bigger than a pre-specified threshold. However, as the link transmission capacity increases, it is unable to process each packet in the large DRAM memories at the speed of current backbone links. Although the small SRAM memories are fast enough for per-packet processing, it is unable to store all the per-flow counters because of the large number of concurrent flows in backbone links (may be one million or more). In this paper, we adopt the deterministic 1-out-of-k packet sampling technique which is widely used in today's operational networks, for instance, it has been implemented in Cisco routers (NetFlow [2]). After sampling, both the packet arriving rates and number of flows are reduced. Consequently, it becomes possible to update the per-flow counters in the large DRAM memories at the speed of backbone links or to store all the per-flow counters in the small SRAM memories. However, since the sampled packets are only a part of the whole packets transmitted, it is critically important to identify high-rate flows correctly. There are several requirements for a good high-rate flow identification algorithm. First of all, the identification accuracy should be satisfied. There should be a low false-positive rate (FPR) and a low false-negative rate (FNR), i.e., a low probability of non-high-rate flows being incorrectly identified and a low probability of high-rate flows being incorrectly not identified, respectively. Second, the operations performed should be very simple, otherwise it will not be suitable for real-time processing. Third, it should be memory efficient as keeping the memory requirement low leads to ease of the implementation. Fourth, it should identify high-rate flows quickly. This is because that the traffic characteristics will change over time, therefore the identification should be accomplished before the traffic varies. Moreover, fast identification is also very important for early detection of DDoS attacks.

Manuscript received May 22, 2009.
Manuscript revised December 16, 2009.

[†]The authors are with Research Center of Computer Network and Information Security Technology, Harbin Institute of Technology, 150001, China.
[††]The authors are with Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100080, China.
a) E-mail: yuzhanghit@gmail.com
b) E-mail: bxfang@ict.ac.cn
c) E-mail: luohao@software.ict.ac.cn
DOI: 10.1587/transinf.E93.D.1162

The rest of this paper is organized as follows. In Sect. 2, we describe relative work before formally defining the high-rate flow identification problem in Sect. 3. We first propose a high-rate flow identification method based on fixed sample size test (FSST) in Sect. 4 and then propose the second novel high-rate flow identification method based on truncated sequential probability ratio test (TSPRT) in Sect. 5. A theoretical analysis of the identification accuracies for both methods is provided in Sect. 6. We present the results of our experimental evaluation in Sect. 7 and the conclusion in Sect. 8.

2. Related Work

In identifying high-rate flows, the identification accuracy is evaluated and determined by the identification curve which gives the identification probability of flows with arbitrary flow rates, e.g., the identification probability of f^* (the flow-rate threshold defining high-rate flows). Therefore, it is practically important that we can specify the identification curve according to the accuracy requirement.

Stabilized RED (SRED) [3] presented a method for identifying high-rate flows in a bottleneck link. The basic idea is to compare the arriving packet with a randomly selected flow from a flow table, SRED increases the packet counter value of the flow if the comparison is successful. High-rate flows are more likely to get higher counter values since they send more packets. Stochastic Fair Blue (SFB) [4] uses L independent hash functions to increase the counter values of the corresponding bins at L levels upon each packet arrival. SFB identifies a flow if all the counters associated with this flow go above a preset threshold. I. Smitha and A. Reddy [5] suggest a method to identify high-rate flows based on LRU cache replacement policy. Random Early Detection with Preferential Dropping (RED-PD) [6] counts the number of packets dropped by RED and identifies flows if the number exceeds a preset threshold during a specified time interval. However, RED-PD can only be used with RED equipped routers in heavy traffic loads. At last, unfortunately there is no way to specify or evaluate the identification accuracy in the above four methods.

According to [1], since there is a strong positive correlation between the flow size and flow rate, the high-rate flows might be identified by using a method that identifies the elephant flows [7]–[12]. However, because the relationship between these two properties can not be formulated, there is no way to specify or evaluate the identification accuracy either.

The short timeout method (ST) [13], [14] is the state-of-the-art mechanism for identifying high-rate flows by using sampled packets, which simply identifies flows from which a fixed number of packets are sampled during the measurement period. Although ST derives the identification probability for flows with arbitrary rates, it can not specify the identification curve according to the accuracy requirement. As a result, we first propose FSST which changes the parameter setting process of ST so that it can identify high-rate flows with user-specified identification accuracy. However, FSST is not memory efficient as it has to record every sampled flow in the flow table during the measurement period. Therefore, we propose TSPRT which can reduce the memory cost and shorten the identification time a lot while satisfying the accuracy requirement as compared to ST.

3. Problem Definition and Dataset

The definition of flows can be very flexible, here we use the definition based on the 5-tuples in the IP header, i.e., all packets that have the same source IP address, destination IP address, source port number, destination port number and protocol identifier are considered to be in the same flow. Let's assume that each arriving packet belongs to one of M flows. During a measurement period Δ of duration δ (seconds), the arriving rate of flow $i \in M$ is defined as: $f_i = \frac{u_i}{\delta}$ (packets/s), where u_i is the number of unsampled packets arriving within Δ from flow i. The flows with $f \geq f^*$ are defined to be high-rate flows, where f^* is an arbitrary threshold. And the objective is *to identify high-rate flows of which the* $f \geq f^*$ *within* Δ.

However, N. Kamiyama and T. Mori [13], [14] suggest setting the traffic ratio threshold p^* directly instead of f^*. Let Λ denote the packet arrival rate in the target link (packets/s) and let's assume both Λ and f_i are constant within Δ [13]–[16]. Let $p_i = \frac{f_i}{\Lambda}$ denote the traffic ratio of flow i, therefore p_i is stationary within Δ [13]–[16], accordingly $p^* = \frac{f^*}{\Lambda}$. Since it is easy to measure Λ, instead of directly identifying high-rate flows of which the $f \geq f^*$, we solve the equivalent problem of identifying high-rate flows of which the $p \geq p^*$. Consequently the objective becomes *to identify high-rate flows of which the* $p \geq p^*$ *within* Δ.

Since the flow rate might vary during a long time, the measurement period Δ should be very short. And if we need to identify high-rate flows during a long time, we can divide the time into several measurement periods, and then identify high-rate flows in each continuous measurement period.

We describe below how to model the high-rate flow identification problem. Over the whole measurement period Δ, let $N = \Lambda \times \delta$ denote the total number of unsampled packets sent by all flows. First let's assume that all packets are sampled with the same sampling interval K, thus the total number of sampled packets $n = \frac{N}{K}$. Then according to [17], the probability of an arrival packet belonging to a given flow is independent of all other packets. This is because that in a high speed backbone link, the number of simultaneous flows is very large and the packets of different flows are highly interleaved, hence the consecutive packets of a given flow are separated by a random number of packets of other flows. Therefore the number of sampled packets belonging to flow i is an approximate *hypergeometric* random variable, the probability of taking value c_i is

$$P\{X = c_i\} = \frac{\binom{u_i}{c_i}\binom{N-u_i}{n-c_i}}{\binom{N}{n}} \tag{1}$$

where $N = \sum_{i=1}^{M} u_i$, $n = \sum_{i=1}^{M} c_i$.

When n is small relative to N (e.g., $K \geq 10$), we can approximate the hypergeometric probabilities with binomial probabilities as below:

$$P\{X = c_i\} = \binom{n}{c_i} p_i^{c_i} (1 - p_i)^{n - c_i} \quad (2)$$

where $p_i = \frac{u_i}{N}$ is the ratio of the number of packets of flow i over the total number of packets during Δ. Because $u_i = f_i \times \delta$, $N = \Lambda \times \delta$, thus $p_i = \frac{u_i}{N} = \frac{f_i \times \delta}{\Lambda \times \delta} = \frac{f_i}{\Lambda}$ is the traffic ratio of flow i. Therefore, the number of sampled packets belonging to flow i is an approximate binomial random variable and we can consider traffic ratio p_i as the probability of an arrival sampled packet belonging to flow i in a Bernoulli trial. Hence, we can call p_i the success probability of flow i as well. *As a result, we can use the number of sampled packets c_i to infer whether p_i is bigger than p^* or not.* At last, the problem becomes to test the hypothesis below:

$H_0 : p < p^*$, *against the alternative*
$H_1 : p \geq p^*$

The four metrics along which we measure the performance of the method are:

- *Identification Accuracy*: We use FPR and FNR to measure the identification accuracy. FPR is defined as the proportion of non-high-rate flows that are incorrectly misidentified. FNR is defined as the proportion of high-rate flows that are incorrectly not identified.
- *Identification Time*: Identification time is the time needed for identifying the high-rate flows with satisfied accuracy. It is different from the measurement time. Since the high-rate flows can be identified before the end of the measurement period, the identification time is no more than the measurement time. Since the traffic characteristics will change over time, we would like the identification time scale to be smaller than the time scale in which the traffic varies.
- *Memory Cost*: We use the number of flows needed to be maintained in memory as surrogate for the amount of memory required. In practice, low memory requirement will lead to ease of the implementation.
- *Sample Size*: Sample size is the number of total sampled packets needed to achieve the desired identification accuracy. Given the same measurement time, the less the sample size is, the bigger the sampling interval K would be, and thus the more the processing cost would be reduced.

For illustration reason, we use the Abilene-III Internet trace data which is measured at an OC192c backbone link by the PMA project of NLANR [18]. It is the first publicly available 10 Gigabit Internet backbone trace. In this trace, we consider the first 2.0×10^7 packets from Indianapolis (IPLS) to Kansas City (KSCY) as in [14], which corresponds to about 147 seconds of the observed traffic. The total number of flows in this truncated trace was 706,571, the average link utilization was 0.207 and the maximum flow rate was 1.75×10^4 packets/s [14]. For convenience, Table 1 summarizes the notations introduced in this paper.

Table 1 Meaning of selected symbols.

Symbol	Meaning
Δ	measurement period
δ	measurement time (seconds)
u_i	number of packets of flow i
f_i	packet arrival rate of flow i
Λ	total packet arrival rate
p	traffic ratio
p_i	traffic ratio of flow i
f^*	packet arrival rate threshold
p^*	traffic ratio threshold
N	total number of packets
K	sampling interval
n	total number of sampled packets
c_i	number of sampled packets of flow i
p_0	limiting high-rate level
p_1	acceptable high-rate level
α	Prob. of misidentifying a flow with p_0
β	Prob. of not identifying a flow with p_1
$I(p)$	Prob. of identifying a flow with p
n^*	sample size
c^*	acceptance number
ε	identification time
m	memory cost
s	slope
h_1	intercept of rejection line
h_2	intercept of acceptance line
m_1, m_2	mixture degree in TSPRT

4. Single Sampling Method

Sampling inspection [19] is an important aspect of statistical quality control. It involves testing a batch of data to determine if the proportion of units having a particular attribute exceeds a given percentage. In identifying high-rate flows, we can use sampling inspection to determine if the traffic ratio of a given flow exceeds a pre-defined threshold. Single sampling plan [19] is the most common and simple sampling plan of which the sample size is fixed at the beginning of the experiment, therefore it is also named fixed sample size test.

4.1 Fixed Sample Size Test

A fixed sample size test (FSST) [19] for attributes denoted as (n, c) consists of a sample of size n and an acceptance number c. The procedure works as follows: first randomly select n items from the lot, and then check the number of defectives. If it is less than c, then the lot is accepted, otherwise the lot is rejected. In a FSST, there are 4 key parameters involved: p_0 (acceptable quality level), p_1 (limiting quality level), α (Type I error probability), β (Type II error probability). The acceptable quality level p_0 is the percent defective that is the baseline requirement for the quality of the product, which means the product is considered to be high quality if its unqualified rate $p \leq p_0$. The limiting quality level p_1 ($p_1 > p_0$) is a designated high defect level that will be unacceptable to a consumer, which means the product is

considered to be low quality if its defective rate $p \geq p_1$. The Type I error probability α is the probability of rejecting a lot that has a defect level $p = p_0$. The Type II error probability β is the probability of accepting a lot that has a defect level $p = p_1$.

The operating characteristic (OC) [19] curve is used to evaluate the performance of a given FSST. The OC curve plots the probability of accepting the lot versus the lot fraction defective and displays the discriminatory power of the sampling plan. For a given FSST (n, c), the probability of finding c or fewer defectives in a sample of size n can be approximated by the binomial distribution. Let X denote the binomial random variable, $L(p)$ denote the probability of accepting a lot having the defect level p, then

$$L(p) = P\{X \leq c\} = \sum_{X=0}^{c} \binom{n}{X} p^X (1-p)^{n-X} \tag{3}$$

From the above equation we can see that for a given FSST (n, c), the probability of acceptance depends upon p which is the actual and unknown proportion of defectives in the lot. Thus, the OC curve can be drawn that gives the probability of accepting a lot as a function of the defective rate p.

Suppose we want to construct a FSST (n, c) such that the Type I error probability is α for lots with fraction defective p_0, and the Type II error probability is β for lots with fraction defective p_1. The sample size n and acceptance number c are the solution to

$$\begin{cases} L(p_0) = 1 - \alpha \\ L(p_1) = \beta \end{cases} \tag{4}$$

4.2 Identifying High-Rate Flows Based on FSST

In order to identify high-rate flows based on FSST, we need to introduce two definitions about the traffic ratio p.

Definition 1 (Limiting High-rate Level p_0): It is a designated low traffic ratio and it would be unacceptable if a flow with this traffic ratio is identified as a high-rate flow, which means the flow is considered to be *low-rate* if its traffic ratio $p \leq p_0$.

Definition 2 (Acceptable High-rate Level $p_1 > p_0$): It is the traffic ratio that is the baseline requirement for the high-rate flows, which means the flow is considered to be *high-rate* if its traffic ratio $p \geq p_1$ and *non-high-rate* if its traffic ratio $p < p_1$. Hence, $p_1 = p^*$ here.

As a result, the problem of identifying high-rate flows becomes to test the hypothesis below:

$$H_0 : p \leq p_0, \text{ against the alternative}$$
$$H_1 : p \geq p_1$$

Accordingly, the Type I error probability α is the probability of misidentifying a flow with traffic ratio $p = p_0$. The Type II error probability β is the probability of not identifying a flow with traffic ratio $p = p_1$.

For a given FSST $(n, c + 1)$, let $I(p)$ denote the probability of identifying a flow with traffic ratio p. Since the number of sampled packets belonging to a flow is an approximate binomial variable (Sect. 3), I(p) can be derived as below:

$$\begin{aligned} I(p) &= P\{X > c\} = \sum_{X=c+1}^{n} \binom{n}{X} p^X (1-p)^{n-X} \\ &= 1 - L(p) \end{aligned} \tag{5}$$

The identification curve of this FSST $(n, c+1)$ is determined by the above equation. Therefore, if we want to construct a FSST $(n, c+1)$ such that the Type I error probability is α for flows with traffic ratio p_0, and the Type II error probability is β for flows with traffic ratio p_1, and then the sample size n and acceptance number c are the solution to

$$\begin{cases} I(p_0) = 1 - L(p_0) = \alpha \\ I(p_1) = 1 - L(p_1) = 1 - \beta \end{cases} \tag{6}$$

It is equivalent to

$$\begin{cases} L(p_0) = 1 - \alpha \\ L(p_1) = \beta \end{cases} \tag{7}$$

It is difficult to get the accurate values of n and c from the above equations, however we can use the *central limit theorem* [20] to estimate these two values. Let $X \sim B(n, p)$ (that is X follows the binomial distribution with parameters n and p), then the expected value of X is $E(X) = np$ and the variance is $Var(X) = np(1 - p)$. Since $B(n, p)$ is the sum of n independent, identically distributed Bernoulli random variables with parameter p, $Z = (X - E(X))/Var(X) = (X - np)/\sqrt{np(1-p)}$ is, for large n, approximately a standard *normal* random variable according to the *theorem of de Moivre-Laplace* [20] which is a special case of the *central limit theorem*. As a result,

$$\begin{cases} L(p_0) = P\{X \leq c\} = P\left\{\dfrac{X - np_0}{\sqrt{np_0q_0}} \leq \dfrac{c - np_0}{\sqrt{np_0q_0}}\right\} = 1 - \alpha \\ L(p_1) = P\{X \leq c\} = P\left\{\dfrac{X - np_1}{\sqrt{np_1q_1}} \leq \dfrac{c - np_1}{\sqrt{np_1q_1}}\right\} = \beta \end{cases}$$

where $q_0 = 1 - p_0$, $q_1 = 1 - p_1$. Obviously, the sample size n and acceptance number c are the solution to

$$\begin{cases} \dfrac{c - np_0}{\sqrt{np_0q_0}} = -Z_\alpha \\ \dfrac{c - np_1}{\sqrt{np_1q_1}} = Z_\beta \end{cases} \tag{8}$$

where Z_α is the α percentile for the unit normal distribution, i.e., $P\{Z \leq Z_\alpha\} = \alpha$, $P\{Z \leq Z_\beta\} = \beta$. By solving the above equations, we get

$$\begin{cases} n^* = \left(\dfrac{Z_\alpha \sqrt{p_0 q_0} + Z_\beta \sqrt{p_1 q_1}}{p_1 - p_0} \right)^2 \\ c^* = \dfrac{p_0 p_1 (Z_\alpha^2 q_0 + Z_\beta^2 q_1) + Z_\alpha Z_\beta \sqrt{p_0 p_1 q_0 q_1} (p_0 + p_1)}{(p_1 - p_0)^2} \end{cases} \tag{9}$$

4.3 Complete Description of FSST

Assume that we are currently in the measurement period Δ_d of duration δ seconds, where $d = 1, 2, \ldots$, then the sampling interval is

$$K = \frac{N}{n^*} \tag{10}$$

where $N = N_d = \Lambda \times \delta$ is the total number of packets in Δ_d. However, N_d is unknown when the identification process starts. Generally, we can use the value of N in the x-th most recent measurement period as in [14], i.e., $N = N_{d-x}$. In practice, we use the value of N in the previous measurement period for convenience, i.e., $N = N_{d-1}$.

Given a FSST $(n^*, c^* + 1)$, let e denote the number of sampled packets needed to identify a high-rate flow with traffic ratio p. It is a negative binomial random variable:

$$P\{e = t|p\} = \binom{t-1}{c^*} p^{c^*+1} (1-p)^{t-c^*-1} \tag{11}$$

where $t = c^* + 1, c^* + 2, \ldots, n^*$. Therefore, the expected number of sampled packets needed to identify a high-rate flow with traffic ratio p is

$$E(e|p) \approx \frac{c^* + 1}{p} \tag{12}$$

Consequently, the expected identification time needed to identify a high-rate flow with traffic ratio p is

$$\varepsilon(p) = E(e|p) \times \frac{K}{\Lambda} \approx \frac{c^* + 1}{p} \times \frac{K}{\Lambda} = \frac{c^* + 1}{p n^*} \times \delta \tag{13}$$

where $\frac{K}{\Lambda} = \frac{\delta}{n^*}$ is derived from Eq. (10). From Eq. (13) we can see that the expected identification time will be less than or equal to $\frac{c^*+1}{p^* n^*} \times \delta$ as a result of $p \geq p^*$.

The memory cost of a given FSST $(n^*, c^* + 1)$ is the memory amount required for the flow table which records every sampled flow during the measurement period. Figure 1 plots the relationship between the average flow count and sample size n^* with standard deviation for three different values of K for the truncated Abilene-III trace. As shown in Fig. 1, the average flow count increases almost linearly with n^* and the bigger the sampling interval K is, the faster the average flow count increases. Therefore, the approximate memory cost m is

$$m \approx r n^* \tag{14}$$

where $0 < r \leq 1$ is a constant coefficient determined by K.

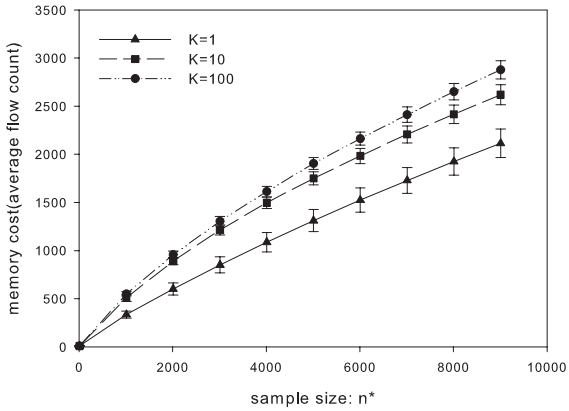


Fig. 1 Memory cost versus sample size n^*.

INPUT
Type I error probability α for flows with limiting high-rate level p_0
Type II error probability β for flows with acceptable high-rate level p_1
Measurement time δ

INITIALIZATION
Compute the sample size
$n^* = \left(\frac{Z_\alpha \sqrt{p_0 q_0} + Z_\beta \sqrt{p_1 q_1}}{p_0 - p_1}\right)^2$
Compute the acceptable number
$c^* = \frac{p_0 p_1 (Z_\alpha^2 q_0 + Z_\beta^2 q_1) + Z_\alpha Z_\beta \sqrt{p_0 p_1 q_0 q_1} (p_0 + p_1)}{(p_0 - p_1)^2}$
where $q_0 = 1 - p_0$, $q_1 = 1 - p_1$ and Z_α is the α percentile for the unit normal distribution.

PROCESSING AT EACH Δ_d
Flow Table FT $\leftarrow \emptyset$
$N = N_{d-1}$
$K = \frac{N}{n^*}$

PROCESSING AT EACH ARRIVAL
When n-th ($n \leq n^*$) packet is sampled
Flow id of current arrival is i
If $i \notin$ FT then
Find empty space
Create flow entry i in FT and set $c_i = 1$
Else
Increment c_i
If $c_i \geq c^* + 1$ then
Identify flow i as high-rate flow

Fig. 2 Identifying high-rate flows based on FSST.

Finally, the overall FSST based high-rate flow identification algorithm is presented in Fig. 2. First of all, we need to compute the sample size n^* and acceptable number c^* according to the input parameters (p_0, p_1, α, β). At the beginning of each measurement period Δ_d, firstly we need to initialize the flow table which stores the flow id i as well as its corresponding number of sampled packets c_i. And then the sampling interval K is computed using the total number of packets in the previous measurement period N_{d-1}. When the n-th ($n \leq n^*$) packet is sampled, we first check whether the flow to which the packet belongs is already in the flow table. If the flow is not in the flow table, then a new entry with $c_i = 1$ is created. Otherwise, we increment the appropriate counter and identify the flow as a high-rate flow if $c_i \geq c^* + 1$. As illustrated in Fig. 2, we can see that the identifying process in each Δ_d will not stop until the total

number of sampled packets reaches n^*.

From the above analysis we can see that although FSST can adjust and bound the identification accuracy through the input parameters α and β, it is not memory efficient due to recording every sampled flow in the table during the measurement period. Is it possible to recognize and remove the low-rate flows at the early stage so that we do not have to keep counters for all the flows during the measurement period? We propose TSPRT below.

5. Sequential Sampling Method

Sequential sampling plan [19] is a special sampling plan of which the sample size is not fixed at the beginning of the experiment. In classical sampling plans, the sample is collected without analysis and consideration. However, in a sequential sampling plan the data is evaluated as it is sampled, and the decision of whether or not needing further sampling depends on the samples observed previously. As a result, the conclusion may be drawn earlier than other classical sampling plans (e.g., FSST). Consequently, the average sample size can be reduced. Theoretically, the sequential sampling may continue infinitely; however, in practice it is often truncated after a certain number of samples.

5.1 Sequential Probability Ratio Test

Sequential probability ratio test (SPRT) methodology proposed by Wald [21] is one kind of the sequential sampling plan, which can satisfy the identification accuracy while requiring the minimum average sample size. Let X denote the number of sampled packets of the inspected flow which is an approximate binomial random variable (See Sect. 3). After n ($n \geq 1$) total packets are sampled, let d_n denote the actual number of sampled packets of the inspected flow, then the probability ratio L_n is

$$L_n = \frac{P\{X = d_n|H_1\}}{P\{X = d_n|H_0\}} = \frac{p_1^{d_n}(1-p_1)^{n-d_n}}{p_0^{d_n}(1-p_0)^{n-d_n}} \tag{15}$$

See definitions of H_0, H_1, p_0, p_1 in Sect. 4.2. Then L_n is compared with two positive constants A and B ($0 < A < 1 < B$). If $L_n \geq B$, this means that there is strong enough statistical evidence to accept hypothesis $H_1 : p \geq p_1$, in other words, the inspected flow is considered to be a high-rate flow. If $L_n \leq A$, this means that there is strong enough statistical evidence to accept the hypothesis $H_0 : p \leq p_0$, that is to say, the inspected flow is considered to be a low-rate flow. Otherwise, continue with more samples. The constants A and B are approximated by the following formulas:

$$A \approx \frac{\beta}{1-\alpha} \tag{16}$$

and

$$B \approx \frac{1-\beta}{\alpha} \tag{17}$$

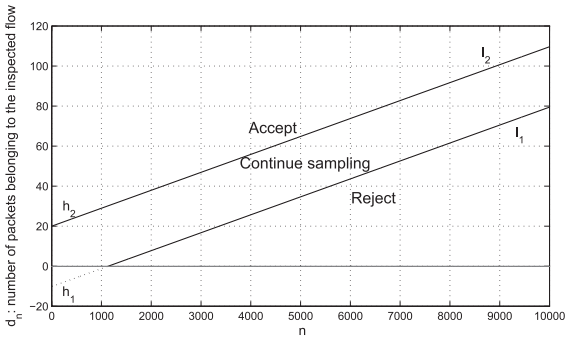


Fig. 3 Graphical illustration of SPRT.

See definitions of α, β in Sect. 4.2. Equivalently, the execution process of SPRT can be described as follows:

$$\begin{cases} \textit{reject } H_1 \ (\textit{accept } H_0) & \text{if } d_n \leq h_1 + n \cdot s \\ \textit{accept } H_1 \ (\textit{reject } H_0) & \text{if } d_n \geq h_2 + n \cdot s \\ \textit{continue sampling} & \text{Otherwise} \end{cases} \tag{18}$$

where $h_1 = \frac{\ln A}{k}$, $h_2 = \frac{\ln B}{k}$, $s = \frac{\ln \frac{1-p_0}{1-p_1}}{k}$, $k = \ln \frac{p_1(1-p_0)}{p_0(1-p_1)}$.

The operation of SPRT can be depicted as in Fig. 3, where the number of arrivals (total number of sampled packets so far), n, is the abscissa, and the number of sampled packets of the inspected flow, d_n, is the ordinate. Two lines are defined as below:

$$l_1 : \ d_n = h_1 + n \cdot s \ \text{ (rejection line)} \tag{19}$$

$$l_2 : \ d_n = h_2 + n \cdot s \ \text{ (acceptance line)} \tag{20}$$

It is obvious that d_n is a linear function of n since h_1, h_2 and s are already determined by the input parameters (p_0, p_1, α, β). Among them, s is the slope of both lines, h_1 ($h_1 < 0$) is the intercept of the rejection line and h_2 ($h_2 > 0$) is the intercept of the acceptance line. According to the execution process of SPRT (Eq. (18)), the accepting region, rejecting region and continue sampling region are bounded by line l_1 and line l_2 in turn. When sampling the n-th packet, if the point (n, d_n) stays within the accepting region, then the inspected flow will be identified as a high-rate flow; if the point (n, d_n) stays within the rejecting region, then the inspected flow will be identified as a low-rate flow; if the point (n, d_n) stays within the continue sampling region, then no conclusion will be drawn and the $(n+1)$-th sample will be needed.

5.2 Truncated Sequential Probability Ratio Test

Theoretically, SPRT can continue infinitely; however, in practice SPRT is often truncated after a certain number of samples. Let n^* denote the predetermined maximum sample size. As a result, the execution process of truncated sequential probability ratio test (TSPRT) can be described as below:

$$\text{if } n < n^* \begin{cases} \textit{reject } H_1 & \text{if } d_n \leq h_1^* + n \cdot s \\ \textit{accept } H_1 & \text{if } d_n \geq h_2^* + n \cdot s \\ \textit{continue sampling} & \text{Otherwise} \end{cases} \tag{21}$$

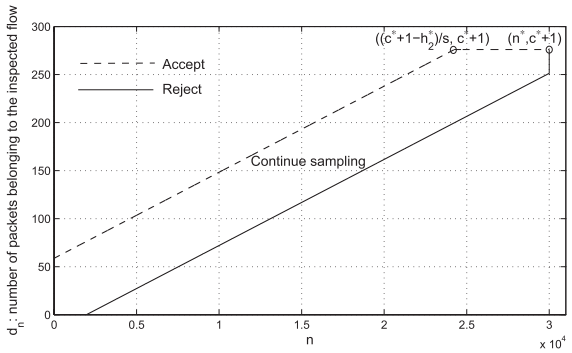


Fig. 4 Graphical illustration of TSPRT.

$$\text{if } n = n^* \begin{cases} \textit{reject } H_1 & \text{if } d_n < c^* + 1 \\ \textit{accept } H_1 & \text{if } d_n \geq c^* + 1 \end{cases} \tag{22}$$

where c^*+1 is the fixed threshold. Therefore, the test will be truncated at $n = n^*$ if it has not been terminated previously.

Figure 4 shows a TSPRT diagram. When sampling the n-th packet, the inspected flow will be identified as a high-rate flow if the point (n, d_n) crosses the acceptance line and regarded as a low-rate flow if the point (n, d_n) crosses the rejection line, otherwise another sample will be needed. A careful reader might notice that the execution process depicted in Fig. 4 is slightly different from the execution process described in Eq. (21) and Eq. (22). The acceptance line becomes a line parallel to the x-axis after reaching the point $(\frac{c^*+1-h_2^*}{s}, c^* + 1)$. This is because that each flow i remaining in the flow table with counter $c_i \geq c^* + 1$ will be identified as a high-rate flow at the end of each measurement period according to Eq. (22). Therefore, the flow is identified as a high-rate flow as soon as its counter $\geq c^* + 1$ which will also accelerate the identification of high-rate flows.

However, the problem is that we need to choose an appropriate n^* so that the identification accuracy of TSPRT will be satisfied. According to [22], by viewing the TSPRT as a mixture of SPRT and FSST, the truncation point (n^*) and boundaries (c^*, h_1^*, h_2^*) can be obtained while satisfying the accuracy requirement (p_0, p_1, α and β) as long as the mixture degree is specified.

Let m_1 and m_2 ($0 < m1, m2 < 1$) be two constants determining the degree of mixture. According to the design procedure proposed in [22], the truncation point and boundaries are

$$\begin{cases} h_1^* = \frac{\ln A^*}{k} \\ h_2^* = \frac{\ln B^*}{k} \\ n^* = \left(\frac{Z_{\alpha_{FSST}} \sqrt{p_0 q_0} + Z_{\beta_{FSST}} \sqrt{p_1 q_1}}{p_0 - p_1} \right)^2 \\ c^* = \\ \quad \frac{p_0 p_1 (Z^2_{\alpha_{FSST}} q_0 + Z^2_{\beta_{FSST}} q_1) + Z_{\alpha_{FSST}} Z_{\beta_{FSST}} \sqrt{p_0 p_1 q_0 q_1} (p_0 + p_1)}{(p_0 - p_1)^2} \end{cases} \tag{23}$$

where $A^* = \frac{\beta_{SPRT}}{1-\alpha_{SPRT}}$, $B^* = \frac{1-\beta_{SPRT}}{\alpha_{SPRT}}$, $k = \ln \frac{p_1(1-p_0)}{p_0(1-p_1)}$, $\alpha_{SPRT} = (1 - m_1)\alpha$, $\beta_{SPRT} = (1 - m_2)\beta$, $\alpha_{FSST} = m_1\alpha$, $\beta_{FSST} = m_2\beta$ and $s = \frac{\ln \frac{1-p_0}{1-p_1}}{k}$. According to Eq. (9) and Eq. (23), we can see that when given the same accuracy constraint (p_0, p_1, α, β), the maximum sample size of TSPRT must be bigger than the sample size of FSST as a result of $\alpha_{FSST} < \alpha$ and $\beta_{FSST} < \beta$. However, in practice we usually first determine the maximum sample size n^* from the hardware limits, and then choose the appropriate mixture degree constants m_1, m_2. Therefore, we propose two ways to choose appropriate m_1, m_2 below.

Unlike FSST, TSPRT based high-rate flow identification algorithm does not need to record every sampled flow during the measurement period. According to the execution process of TSPRT (Eq. (21)), we can see that the inspected flow is considered to be a low-rate flow if its number of sampled packets d_n is less than or equal to $h_1^* + n \cdot s$. As a result, its record in the flow table can be removed or reused for new arriving flows. Therefore, the memory cost of TSPRT will be much less than that of FSST. In the formula $h_1^* + n \cdot s$, n is a variable which records the total number of sampled packets, s is determined by p_0 and p_1, only h_1^* is involved with m_1 and m_2. From the above analysis, we can see that the bigger the parameter h_1^* is, the earlier the low-rate flows will be removed from the flow table, thus the memory cost will be reduced. As a result, m_1 and m_2 can be chosen to maximize h_1^* which leads to a TSPRT with minimum memory cost.

Let TSPRT-M denote the TSPRT of which the mixture degree constants m_1 and m_2 are chosen to maximize h_1^*. However, the side effect is that the resultant parameters (h_1^*, h_2^*) favor FPR over FNR, which means that FPR will be reduced as much as possible. This is because the low-rate flows will be removed as early as possible, it will be of course more difficult for a low-rate flow to be misidentified which leads to a smaller FPR. Since there is a tradeoff between FPR and FNR, FNR will be bigger than usual.

There is another way to choose m_1 and m_2. According to the execution process of TSPRT (Eq. (21)), the inspected flow is considered to be a high-rate flow if its number of sampled packets d_n is more than or equal to $h_2^* + n \cdot s$. As a result, the less the $h_2^* + n \cdot s$ is, the earlier the high-rate flow will be identified, thus the identification time will be reduced. In the formula $h_2^* + n \cdot s$, only h_2^* is involved with m_1 and m_2. Thereby, m_1 and m_2 can be chosen to minimize h_2^* which leads to a TSPRT with minimum identification time.

Let TSPRT-T denote the TSPRT of which the mixture degree constants m_1 and m_2 are chosen to minimize h_2^*. However, the side effect is that the resultant parameters (h_1^*, h_2^*) favor FNR over FPR, which means that FNR will be reduced as much as possible. This is because that the high-rate flows will be identified as early as possible, it will be of course more difficult to miss a high-rate flow which then leads to a smaller FNR. Since there is a tradeoff between FPR and FNR, FPR will be bigger than usual.

5.3 Complete Description of TSPRT

Assume that we are currently in measurement period Δ_d of duration δ seconds, where $d = 1, 2, \ldots$, then the sampling

interval is

$$K = \frac{N}{n^*} \tag{24}$$

where $N = N_d = \Lambda \times \delta$ is the total number of packets in Δ_d. Since N_d is unknown when the identification process starts, in practice we can use the value of N in the previous measurement period for convenience, i.e., $N = N_{d-1}$.

Let us explore the expected number of sampled packets needed to identify a high-rate flow with traffic ratio p in TSPRT. First, if the inspected flow is identified under the condition of its counter $\geq c^* + 1$ which means in the case of FSST, then the expected number of sampled packets is about $\frac{c^*+1}{p}$ (See Sect. 4.2). Second, if the inspected flow is identified under the condition of its counter $\geq h_2^* + n \cdot s$ which means in the case of SPRT, then according to [23] the expected number of sampled packets is a joint function of $\alpha_{SPRT}, \beta_{SPRT}, p_0, p_1$: $\frac{(1-I(p))h_1^* + I(p)h_2^*}{p-s}$. Among them

$$I(p) = \frac{1 - A^{*h}}{B^{*h} - A^{*h}} \tag{25}$$

See the definitions of h_1^*, h_2^*, s, A^* and B^* in Sect. 5.2. Equation (25) determines the identification probability of a flow with traffic ratio p in SPRT. Since h is determined by $p = \frac{1-(\frac{1-p_1}{1-p_0})^h}{(\frac{p_1}{p_0})^h - (\frac{1-p_1}{1-p_0})^h}$, $I(p)$ is a function of p when given p_0, p_1, α_{SPRT}, β_{SPRT}. Moreover, $I(p_0) = \alpha_{SPRT}$ and $I(p_1) = 1 - \beta_{SPRT}$. Let e denote the expected number of sampled packets needed to identify a high-rate flow with traffic ratio p in TSPRT, then it is

$$E(e|p) = \mathbf{min}\left(\frac{c^* + 1}{p}, \frac{(1 - I(p))h_1^* + I(p)h_2^*}{p - s}, n^*\right) \tag{26}$$

Moreover, when $p = p_1$, the expected number of sampled packets in SPRT is $\frac{\beta_{SPRT}h_1^* + (1-\beta_{SPRT})h_2^*}{p_1 - s}$. Since $p_1 = p^*$, the upper bound of expected number of sampled packets in TSPRT is

$$E(e|p) \leq \mathbf{min}\left(\frac{c^* + 1}{p^*}, \frac{\beta_{SPRT}h_1^* + (1 - \beta_{SPRT})h_2^*}{p^* - s}, n^*\right) \tag{27}$$

where $p \geq p^*$.

The expected identification time and its upper bound in TSPRT are derived as below:

$$\varepsilon(p) = E(e|p) \times \frac{K}{\Lambda} = E(e|p) \times \frac{\delta}{n^*} \tag{28}$$

where $\frac{K}{\Lambda} = \frac{\delta}{n^*}$ is derived from Eq. (24).

$$\varepsilon(p) \leq \mathbf{min}\left(\frac{c^* + 1}{p^*}, \frac{\beta_{SPRT}h_1^* + (1 - \beta_{SPRT})h_2^*}{p^* - s}, n^*\right) \times \frac{\delta}{n^*} \tag{29}$$

where $p \geq p^*$.

INPUT
- Type I error probability α for flows with limiting high-rate level p_0
- Type II error probability β for flows with acceptable high-rate level p_1
- Measurement time δ

INITIALIZATION
- Compute the parameter s from p_0 and p_1
- Determine the sample size n^* from hardware limits
- Choose the appropriate m_1, m_2 to maximize h_1^* or minimize h_2^*
- Compute the parameters h_1^*, h_2^*, c^* from $m_1, m_2, p_0, p_1, \alpha, \beta$

PROCESSING AT EACH Δ_d
- Flow Table FT $\leftarrow \emptyset$
- $N = N_{d-1}$
- $K = \frac{N}{n^*}$

PROCESSING AT EACH ARRIVAL
When n-th ($n \leq n^*$) packet is sampled
- Flow id of current arrival is i
- If $i \notin$ FT
  - If $n < -\frac{h_1^*}{s}$ then
    - Find empty space or entries with the counter $\leq h_1^* + n \cdot s$
    - Create flow entry i in FT and set $c_i = 1$
  - Else
    - Increment c_i
    - If $c_i \geq c^* + 1$ or $c_i \geq h_2^* + n \cdot s$ then
      - Identify flow i as a high-rate flow
    - Else if $c_i \leq h_1^* + n \cdot s$ then
      - remove entry i

Fig. 5 Identifying high-rate flows based on TSPRT.

Finally, the overall TSPRT based high-rate flow identification algorithm is presented in Fig. 5. First of all, we need to compute the parameters s, h_1^*, h_2^*, c^*. At the beginning of each measurement period Δ_d, firstly we need to initialize the flow table which stores the flow id i as well as its corresponding number of sampled packets c_i. And then the sampling interval K is computed using the total number of packets in the previous measurement period N_{d-1}. There are six points to be noted. First, the mixture degree constants m_1 and m_2 are chosen appropriately according to the user requirement (TSPRT-M or TSPRT-T). Second, the new arriving flows will be recorded in the flow table only when $n < -\frac{h_1^*}{s}$. This is because that if a new flow arrives after $n = -\frac{h_1^*}{s}$ which means the number of sampled packets for this flow is $d_n = 0$, and then according to the execution process of TSPRT (Eq. (21)), this flow is considered to be low-rate. Therefore, we do not need to record this flow. Thus, there will be no new flows to be recorded in the table after $n \geq -\frac{h_1^*}{s}$. As a result, the maximum number of flows recorded in the table is the number of flows in the first $\mathbf{min}(-\frac{h_1^*}{s}, n^*)$ sampled packets during the measurement period. Therefore, the approximate memory cost of TSPRT is

$$m \approx r \cdot \mathbf{min}\left(-\frac{h_1^*}{s}, n^*\right) \tag{30}$$

where $0 < r \leq 1$ is a constant coefficient. Third, since flows with counter $\leq h_1^* + n \cdot s$ are considered to be low-rate, their entries in the flow table can be reused when a new flow entry needs to be created. Fourth, the flow will be identified as a high-rate flow as soon as its counter $\geq$

min$(c^* + 1, h_2^* + n \cdot s)$. Fifth, we do not need to check each flow remaining in the table at the end of the measurement period as compared to the execution process described in Eq. (22). This is because that each flow with counter $\geq c^*+1$ has already been identified and the rest flows remaining in the table all have counters $< c^* + 1$. Last, the flow entry is removed from the table as soon as the flow is consider to be low-rate ($c_i \leq h_1^* + n \cdot s$).

6. Identification Accuracy Analysis

In this section, we give the formal definitions of FPR and FNR, and then investigate the relationship between FPR, FNR and p_0, p_1, α, β. The following discussion applies to both FSST and TSPRT.

Let $I(p)$ denote the identification curve which determines the identification probability of a flow with traffic ratio p. The ideal identification curve is the unit step function as presented in Fig. 6 in which $I(p) = 0$ for $p < p_1$ and $I(p) = 1$ for $p \geq p_1$, where $p_1 = p^* = 0.01$. However, we can not get such a good identification curve unless the sampling interval $K = 1$ which means there will be no packet sampling in identifying high-rate flows. Figure 6 also presents an actual identification curve ($I(p_0) = \alpha$, $I(p_1) = 1 - \beta$). From this figure we can see that the actual identification curve is very close to the ideal identification curve except that it is a steep curve not a straight line around $p = p_1$.

Let us start analyzing what happens when considering only one flow. Let p ($0 < p < 1$) denote the traffic ratio of this flow with a uniform prior distribution, FNR_1 denote the probability of not identifying when it is a high-rate flow, FPR_1 denote the probability of misidentifying when it is not a high-rate flow. They are defined by

$$FNR_1 = \frac{\int_{p_1}^{1}(1 - I(p))dp}{1 - p_1} \tag{31}$$

$$FPR_1 = \frac{\int_{0}^{p_1} I(p)dp}{p_1} \tag{32}$$

where $\int_{p_1}^{1}(1 - I(p))dp$ is the size of area surrounded by the identification curve and the unit step function when $p \geq p_1$ (See S_{FNR_1} in Fig. 6), $\int_{0}^{p_1} I(p)dp$ is the size of area surrounded by the identification curve and the unit step function when $p < p_1$ (See $S_{FPR_1} = S_{FPR_1^{p_0}} + S_{FPR_1^{p_1-p_0}}$ in Fig. 6). From Fig. 6 we can obtain that $S_{FNR_1} \leq (1 - p_1)\beta$, therefore $FNR_1 = \frac{S_{FNR_1}}{1-p_1} \leq \frac{(1-p_1)\beta}{1-p_1} = \beta$.

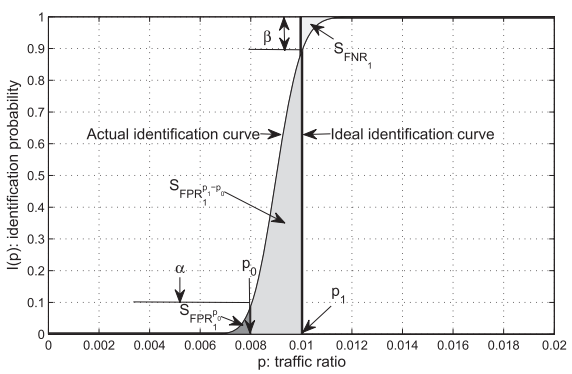


Fig. 6 Ideal and actual identification curve.

Let $FPR_1^{p_0}$ denote the FPR of one given flow with traffic ratio $p \leq p_0$. Then it is defined by

$$FPR_1^{p_0} = \frac{\int_0^{p_0} I(p)dp}{p_0} \tag{33}$$

Since $I(p)$ is a monotone increasing function (For both FSST and TSPRT), $I(0) \leq I(p) \leq I(p_0)$ when $0 \leq p \leq p_0$. Thus, $FPR_1^{p_0} = \frac{\int_0^{p_0} I(p)dp}{p_0} \leq \frac{\int_0^{p_0} I(p_0)dp}{p_0} = I(p_0) = \alpha$. Consequently, $FPR_1 = \frac{\int_0^{p_1} I(p)dp}{p_1} < \frac{\int_0^{p_1} I(p)dp}{p_0} = \frac{\int_0^{p_0} I(p)dp + \int_{p_0}^{p_1} I(p)dp}{p_0} = FPR_1^{p_0} + \frac{\int_{p_0}^{p_1} I(p)dp}{p_0} \leq \alpha + \frac{\int_{p_0}^{p_1} I(p)dp}{p_0}$. From Fig. 6 we can obtain that $\int_{p_0}^{p_1} I(p)dp = S_{FPR_1^{p_1-p_0}} \leq (p_1 - p_0)(1 - \beta)$, therefore $FPR_1 < \alpha + \frac{(p_1-p_0)(1-\beta)}{p_0}$.

However, since the number of simultaneous flows in a high speed link is extremely large (may be more than 10,000 per second in an OC192c backbone link), we shall consider the FNR and FPR of all flows, not just one flow. Let $G(p)$ denote the frequency distribution of traffic ratio for all flows, then FNR and FPR are defined by

$$\begin{aligned} FNR &= \frac{\int_{p_1}^{1}(1 - I(p))G(p)dp}{\int_{p_1}^{1} G(p)dp} \\ &= 1 - \frac{\int_{p_1}^{1} I(p)G(p)dp}{\int_{p_1}^{1} G(p)dp} \end{aligned} \tag{34}$$

$$FPR = \frac{\int_0^{p_1} I(p)G(p)dp}{\int_0^{p_1} G(p)dp} \tag{35}$$

Since $I(p)$ is a monotone increasing function, $I(p_1) \leq I(p) \leq I(1)$ when $p_1 \leq p \leq 1$. Thus, $1 - I(1) \leq 1 - I(p) \leq 1 - I(p_1)$ when $p_1 \leq p \leq 1$. Therefore, $FNR = \frac{\int_{p_1}^{1}(1-I(p))G(p)dp}{\int_{p_1}^{1} G(p)dp} \leq \frac{\int_{p_1}^{1}(1-I(p_1))G(p)dp}{\int_{p_1}^{1} G(p)dp} = (1 - I(p_1))\frac{\int_{p_1}^{1} G(p)dp}{\int_{p_1}^{1} G(p)dp} = 1 - I(p_1) = \beta$. This means that whatever the parameters p_0, p_1, α and β set, $FNR \leq \beta$.

The relative frequency distribution of the traffic ratio for three different values of δ is presented in Fig. 7. In order to get this distribution, we divide the traffic ratio into 10,000 bins between 0 and 1. Since the average relative frequency distribution is extremely small when traffic ratio is bigger than 0.05, we omit the values when $p \geq 0.05$. From Fig. 7 we can clearly see that the frequency distribution of traffic ratio decays in an approximate power-law fashion. As a result, $G(p)$ is an approximate monotone decreasing function.

Suppose $\alpha(t) : [a, b] \to \mathbf{R}$ is monotone increasing, and $f, g : [a, b] \to \mathbf{R}$ are both monotone increasing or decreasing. Then

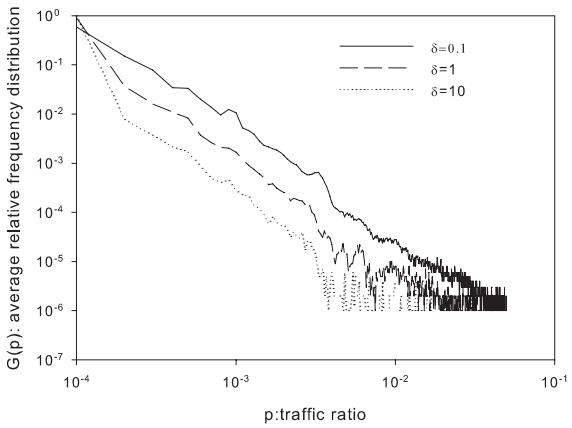


Fig. 7 Average relative frequency distribution of traffic ratio.

$$\left(\int_a^b f d\alpha\right)\left(\int_a^b g d\alpha\right) \le [\alpha(b)-\alpha(a)]\int_a^b f g d\alpha \quad (36)$$

The orientation of the inequality sign changes if f is monotone increasing and g is monotone decreasing. This is the famous Grüss-type inequality. As a result, we have $FPR = \frac{\int_0^{p_1} I(p)G(p)dp}{\int_0^{p_1} G(p)dp} \le \frac{\frac{1}{p_1-0}\int_0^{p_1} I(p)dp \int_0^{p_1} G(p)dp}{\int_0^{p_1} G(p)dp} = \frac{1}{p_1}\int_0^{p_1} I(p)dp = FPR_1$. Since $FPR_1 < \alpha + \frac{(p_1-p_0)(1-\beta)}{p_0}$, we have $FPR < \alpha + \frac{(p_1-p_0)(1-\beta)}{p_0}$. To sum up, we have

$$FPR < \alpha + \frac{(p_1-p_0)(1-\beta)}{p_0} \quad (37)$$

$$FNR \le \beta \quad (38)$$

FNR is less than β whatever the parameters p_0, p_1, α and β set according to Eq. (38). And by carefully selecting the input parameters p_0, p_1, α and β, FPR will not exceed α too much according to Eq. (37). Since both FPR and FNR are proportional to α and β, FPR (FNR) improves as α (β) decreases while other parameters stay unchanged.

7. Evaluation

In this section, we first make a comparison between TSPRT and FSST (as well as ST) in terms of four metrics (identification accuracy, identification time, memory cost and sample size) through an actual packet sampling process for the truncated NLANR trace. Since the maximum sample size n^* is an important parameter governing the behavior of TSPRT, we then explore how n^* affect the resultant performance of TSPRT. At last, we investigate the performance of TSPRT and FSST with different measurement time.

Within the measurement period, let $\hat{n_h}$ denote the number of identified high-rate flows in sampled packets, n_h denote the number of actual high-rate flows among the $\hat{n_h}$ identified high-rate flows, N_h denote the number of total actual high-rate flows in unsampled packets and N_a denote the number of all flows in unsampled packets. As a result, the experimental FPR and FNR can be defined by

$$FPR = \frac{\hat{n}_h - n_h}{N_a - N_h} \quad (39)$$

Table 2 Comparison of TSPRT and FSST.

Algorithm	FPR	FNR	ε(s)	m
TSPRT-M	0.460e-4	0.758e-2	0.911	1289.699
TSPRT-T	0.460e-4	0.758e-2	0.828	1819.000
FSST	0.470e-4	0.675e-2	0.925	2141.671

$$FNR = \frac{N_h - n_h}{N_h} \quad (40)$$

We consider the problem to identify high-rate flows of which $p_i \ge 0.01$ in the truncated NLANR trace under the following accuracy constraint: the probability of misidentifying a flow with $p_i = 0.008$ is less than 0.2 and so is the probability of not identifying a flow with $p_i = 0.01$. That is to say, the parameters are: $p_0 = 0.008$, $p_1 = 0.01$, $\alpha = 0.2$, $\beta = 0.2$.

7.1 Comparison of Methods

So far as we know, ST [13], [14] is the only method which is able to derive the identification probability for flows with arbitrary rates as well as the identification curve which demonstrates the identification accuracy. Therefore, we make a comparison between our proposed methods and ST.

7.1.1 Relationship between FSST and ST

FSST and ST are the same in essence. They all simply identify flows from which a certain number of packets are sampled during the measurement period. The difference is that we put forward the concepts of limiting high-rate level p_0 and acceptable high-rate level p_1 as well as their corresponding Type I error probability α and Type II error probability β in FSST which enable us to specify the identification curve according to the accuracy requirement. Therefore, we use FSST in place of ST when compared with TSPRT.

7.1.2 Comparison of FSST and TSPRT

According to the algorithm of FSST in Fig. 2, we get the following approximate sample size and acceptance number: $n^* = 6297$, $c^* = 56$. We choose the maximum sample size $n^* = 6350$ for TSPRT. Consequently, the parameters of TSPRT-M are: $h_1^* = -27.971$, $h_2^* = 52.052$, $s = 0.896 \times 10^{-2}$, $c^* = 56$ and the parameters of TSPRT-T are: $h_1^* = -44.966$, $h_2^* = 27.560$, $s = 0.896 \times 10^{-2}$, $c^* = 56$. Table 2 shows the comparison result of TSPRT and FSST when $\delta = 2s$. The result is an average of 73 experiments as the truncated NLANR trace contains about $147s$ traffic. There are five points to be noted.

First, we explore the identification accuracies of FSST and TSPRT. As expected, the false positive rates (FPRs) of TSPRT-M, TSPRT-T and FSST are very close to each other, and so are the false negative rates (FNRs). This is because that they have the same accuracy constraint, i.e., $p_0 = 0.008$, $p_1 = 0.01$, $\alpha = 0.2$, $\beta = 0.2$. FPRs are less than $\alpha = 0.2$ and

FNRs are less than $\beta = 0.2$.

Second, we explore the average identification time of FSST and TSPRT. The average identification time of TSPRT-M ($\varepsilon = 0.911$) and TSPRT-T ($\varepsilon = 0.828$) is less than that of FSST ($\varepsilon = 0.925$). This means that TSPRT spends less time in identifying high-rate flows as compared to FSST which is very important for some applications such as DDoS attack detection. This is because that the high-rate flows can be identified before reaching the maximum sample size according to execution process of TSPRT (Eq. (21)). And as expected, the average identification time of TSPRT-T ($\varepsilon = 0.828$) is less than that of TSPRT-M ($\varepsilon = 0.911$). This is because that the mixture degree constants m_1 and m_2 in TSPRT-T are chosen to minimize h_2^* which results in the early identification of high-rate flows.

Third, we explore the memory costs of FSST and TSPRT. As can be seen, both the memory costs of TSPRT-M ($m = 1289.699$) and TSPRT-T ($m = 1819.000$) are less than that of FSST ($m = 2141.671$). This is because that TSPRT only needs to record the sampled flows when $n < -\frac{h_1^*}{s}$; however, FSST has to record every sampled flow during the whole measurement period. And as expected, the memory cost of TSPRT-M ($m = 1289.699$) is less than that of TSPRT-T ($m = 1819.000$). This is because that the mixture degree constants m_1 and m_2 in TSPRT-M are chosen to maximize h_1^* which results in the early removal of low-rate flows.

Fourth, we examine the sample size and sampling interval for both FSST and TSPRT. The maximum sample size of TSPRT ($n^* = 6350$) must be bigger than that of FSST ($n^* = 6297$), otherwise we can not get valid mixture degree constants m_1 and m_2 (See Sect. 5.2). As a result, the sampling interval of TSPRT will be a little smaller than that of FSST with the same measurement time. Therefore, the processing cost of TSPRT is a little bigger than that of FSST.

Last, we explore the operations performed by FSST and TSPRT. As shown in Fig. 2 and Fig. 5, all the operations of FSST and TSPRT are simple in general. Perhaps, the most time-consuming operation is to search the flow table at each packet sampling which can be accomplished by hash techniques. Therefore, both the operations of FSST and TSPRT are very suitable for real-time processing.

7.2 The Effect of n^* in TSPRT

The maximum sample size n^* is an important parameter governing the behavior of TSPRT. However, since we do not change the accuracy constraint (p_0, p_1, α and β), FPR and FNR will not change too much. Thus, we just study the effect of n^* ($6{,}350 \le n^* \le 26{,}000$) on the memory cost m and identification time ε when $\delta = 2s$. Let us take TSPRT-M and TSPRT-T as examples.

The effect of n^* on the identification time with standard deviation is presented in Fig. 8. The identification time of both TSPRT-M and TSPRT-T decreases with n^*. This is because that h_2^* in both TSPRT-M and TSPRT-T decreases with n^*. Since h_2^* is the intercept of acceptance line in Fig. 4, it is obvious that with the decrease of h_2^*, high-rate flows will be identified earlier than usual. Table 3 lists the calculated h_2^* of TSPRT-M. The phase step of h_2^* around $n^* = 20000$ leads to the phase transition of TSPRT-M in Fig. 8. As expected, the identification time of TSPRT-T decreases much faster than that of TSPRT-M.

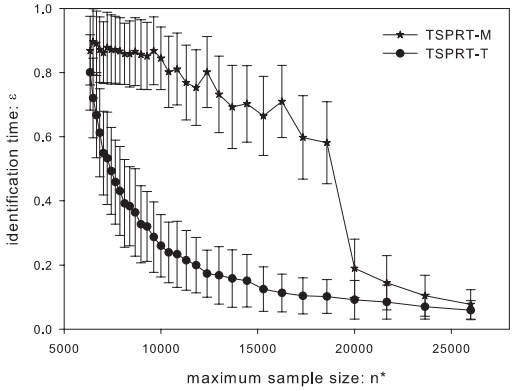


Fig. 8 ε versus n^* of TSPRT-M and TSPRT-T.

Table 3 Calculated h_1^* and h_2^*.

n^*	h_1^* (TSPRT-T)	h_2^* (TSPRT-M)
16250	−44.133	54.903
17333	−44.104	43.440
18571	−21.033	44.007
20000	−14.810	15.806
21666	−12.306	12.684

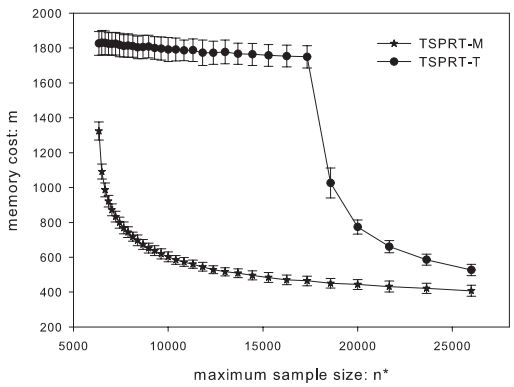


Fig. 9 m versus n^* of TSPRT-M and TSPRT-T.

The effect of n^* on the memory cost with standard deviation is presented in Fig. 9. Both the memory costs of TSPRT-M and TSPRT-T decrease with n^*. This is because that h_1^* ($h_1^* < 0$) in both TSPRT-M and TSPRT-T increases with n^*. According to Eq. (30), it is obvious that the memory cost will be reduced with the increase of h_1^*. Table 3 lists the calculated h_1^* of TSPRT-T. The phase step of h_1^* around $n^* = 18571$ leads to the phase transition of TSPRT-T in Fig. 9 As expected, the memory cost of TSPRT-M decreases much faster than that of TSPRT-T. To sum up, the maximum sample size n^* in TSPRT is involved with the tradeoff

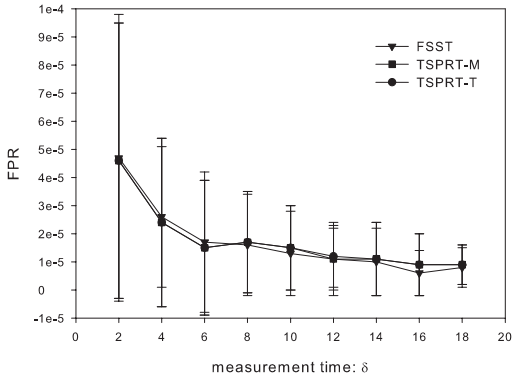


Fig. 10 FPR versus δ of FSST and TSPRT.

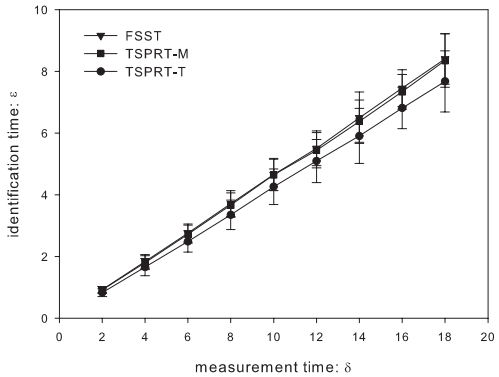


Fig. 12 ε versus δ of FSST and TSPRT.

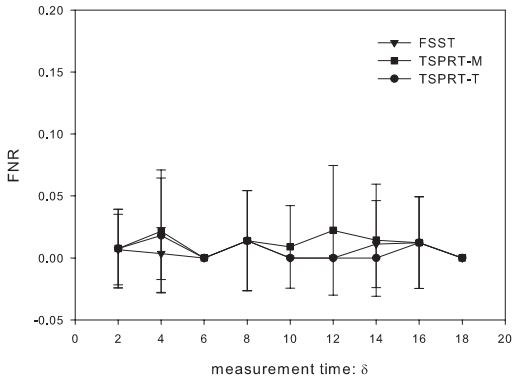


Fig. 11 FNR versus δ of FSST and TSPRT.

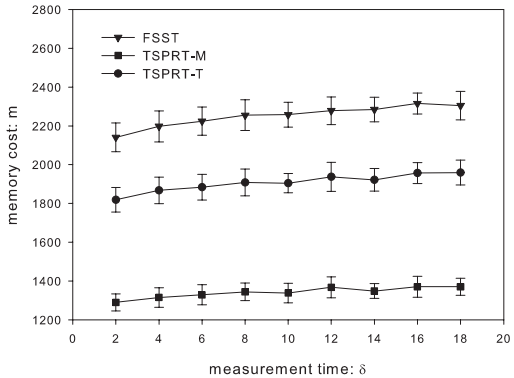


Fig. 13 m versus δ of FSST and TSPRT.

between the processing cost (n^*), memory cost (m) and identification time (ε).

7.3 The Effect of δ in FSST and TSPRT

In this section, we investigate the performance of FSST and TSPRT when the measurement time δ ranges from $2s$ to $18s$. Since the accuracy constraint (p_0, p_1, α and β) stays unchanged, the parameters of FSST and TSPRT remain unchanged too (See Sect. 7.1.2). All the figures are plotted with standard deviation. There are three points to be noted.

First, from Fig. 10 we can see that FPRs of FSST, TSPRT-M and TSPRT-T with different measurement time are very close to each other, so are the FNRs as shown in Fig. 11. This is because that they share the same accuracy constraint.

Second, as shown in Fig. 12, the average identification time of FSST, TSPRT-M and TSPRT-T increases almost linearly with the measurement time. This is because that the expected identification time is proportional to the measurement time when other parameters stay unchanged according to Eq. (13) and Eq. (28). The identification time of FSST and TSPRT-M is very close to each other. As expected, the identification time of TSPRT-T is less than that of FSST and TSPRT-M all along.

Last, the memory costs of FSST, TSPRT-M and TSPRT-T increase slightly with the measurement time as shown in Fig. 13. According to Eq. (10) and Eq. (24), when the sample size maintains unchanged, the longer the measurement time is, the bigger the sampling interval will be which then results in a larger memory cost (See Sect. 4.3). Moreover, both the memory costs of TSPRT-M and TSPRT-T are much less than that of FSST all along.

8. Conclusion and Future Work

In this paper, we study the problem of identifying high-rate flows in high speed backbone links. We proposed two methods: FSST and TSPRT. Compared to the state-of-the-art high-rate flow identification method ST [13], [14], FSST and TSPRT can specify the identification curve according to the accuracy requirement. To be exact, FSST changes the parameter setting process of ST so that it can identify high-rate flows with user-specified identification accuracy. FSST and ST share the same memory cost and identification time. Moreover, TSPRT can decrease the memory cost and identification time by removing the low-rate flows and identifying the high-rate flows at the early stage as compared to ST. We proposed two ways to choose the mixture degree constants m_1 and m_2 in TSPRT: memory cost optimization (TSPRT-M) which is suitable when low memory cost is preferred and identification time optimization (TSPRT-T) which is suitable when short identification time is preferred. Through an actual packet sampling process for the truncated NLANR

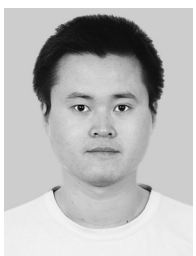
trace, TSPRT was compared to FSST (as well as ST) in terms of four metrics (identification accuracy, identification time, memory cost and sample size). The results showed that with a slightly relaxed processing power, TSPRT required less memory and identification time in identifying high-rate flows with satisfied accuracy as compared to FSST (as well as ST). Although, the identification accuracy can be adjusted by specifying different identification curves, we still can not specify the exact FPR and FNR in FSST or TSPRT. We attempt to address this issue as part of the future work.

Acknowledgment

This study is supported by the National Natural Science Foundation of China (Grant No. 60703021) and the National High-Tech Development 863 Program of China (Grant Nos. 2007AA010501, 2007AA01Z444, 2007AA01Z406, 2009AA012437).

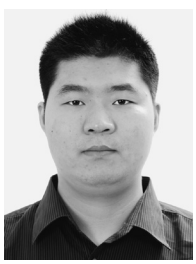
References

[1] Y. Zhang, L. Breslau, V. Paxson, and S. Shenker, "On the characteristics and origins of internet flow rates," Proc. 2002 SIGCOMM Conference, vol.32, no.4, pp.309–322, 2002.
[2] CISCO NetFlow. http://www.cisco.com/en/US/products/ps6601/products_ios_protocol_group_home.html
[3] T. Ott, T. Lakshman, and L. Wong, "SRED: Stabilized RED," INFOCOM'99, Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies, Proc. IEEE, vol.3, 1999.
[4] W. Feng, K. Shin, D. Kandlur, and D. Saha, "The BLUE active queue management algorithms," IEEE/ACM Trans. Netw., vol.10, no.4, pp.513–528, 2002.
[5] I. Smitha and A. Reddy, "Identifying long-term high-bandwidth flows at a router," Proc. 8th International Conference on High Performance Computing, pp.361–371, 2001.
[6] R. Mahajan, S. Floyd, and D. Wetherall, "Controlling high-bandwidth flows at the congested router," Proc. IEEE ICNP'01, 2001.
[7] C. Estan and G. Varghese, "New directions in traffic measurement and accounting: Focusing on the elephants, ignoring the mice," ACM Trans. Comput. Syst., vol.21, no.3, pp.270–313, 2003.
[8] B. Choi, J. Park, and Z. Zhang, "Adaptive packet sampling for accurate and scalable flow measurement," IEEE Global Telecommunications Conference, GLOBECOM'04, 2004.
[9] T. Mori, M. Uchida, R. Kawahara, J. Pan, and S. Goto, "Identifying elephant flows through periodically sampled packets," Proc. 4th ACM SIGCOMM Conference on Internet Measurement, pp.115–120, ACM New York, NY, USA, 2004.
[10] L. Che, B. Qiu, and H. Wu, "Improvement of LRU cache for the detection and control of long-lived high bandwidth flows," Comput. Commun., vol.29, no.1, pp.103–113, 2005.
[11] F. Raspall, S. Sallent, and J. Yufera, "Shared-state sampling," Proc. 6th ACM SIGCOMM Conference on Internet Measurement, pp.1–14, ACM New York, NY, USA, 2006.
[12] F. Raspall and S. Sallent, "Adaptive shared-state sampling," Proc. 8th ACM SIGCOMM Conference on Internet Measurement, pp.271–284, ACM New York, NY, USA, 2008.
[13] N. Kamiyama, "Identifying high-rate flows with less memory," INFOCOM 2005, Proc. IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies, vol.4, 2005.
[14] N. Kamiyama and T. Mori, "Simple and accurate identification of high-rate flows by packet sampling," INFOCOM 2006, Proc. 25th IEEE International Conference on Computer Communications, pp.1–13, 2006.
[15] M. Kodialam, T. Lakshman, and S. Mohanty, "Runs based traffic estimator (RATE): A simple, memory efficient scheme for per-flow rate estimation," INFOCOM 2004, Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies, vol.3, 2004.
[16] F. Hao, M. Kodialam, T. Lakshman, and H. Zhang, "Fast, memory-efficient traffic estimation by coincidence counting," Proc. IEEE INFOCOM, 2005.
[17] Y. Chabchoub, C. Fricker, F. Guillemin, and P. Robert, "Deterministic versus probabilistic packet sampling in the Internet," Lect. Notes Comput. Sci., vol.4516, p.678, 2007.
[18] NLANR: Abilene-III data set. http://pma.nlanr.net/Special/ipls3.html
[19] D. Montgomery, Introduction to statistical quality control, John Wiley & Sons, New York, 2005.
[20] R. Durrett, Probability: Theory and examples, Duxbury Press Belmont, CA, 1996.
[21] A. Wald, "Sequential tests of statistical hypotheses," Annals of Mathematical Statistics, vol.16, no.2, pp.117–186, 1945.
[22] S. Tantaratana and H. Poor, "Asymptotic efficiencies of truncated sequential tests," IEEE Trans. Inf. Theory, vol.28, no.6, pp.911–923, 1982.
[23] A. Wald, Sequential Analysis, John Wiley & Sons, 1947.



Yu Zhang is a Ph.D. Candidate of Research Center of Computer Network and Information Security Technology, Harbin Institute of Technology. His primary research focus lies in network measurement and network security.

Binxing Fang is a professor of Department of Computer Science and Engineering, Harbin Institute of Technology. He is academician of Chinese Academy of Engineering. His research interests include computer network and information security.



Hao Luo is a Ph.D. and Assistant Professor of Research Center of Information Intelligent and Information Security Institute of Computing Technology, Chinese Academy of Sciences His research interest is network security.