International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 1 Issue 5, July - 2012

# A Statistical Loom to Upgrading Dynamic Packet Failure Determination

Shaik Johny Basha[1], R.V. Kishore Kumar[2]

[1]*Assistant Professor, Department of CSE, Universal College of Engineering & Technology,*

[2]*Associate Professor, Department of CSE, Sri Mittapalli College of Engineering,*

## Abstract

*Determination and inference of packet failure features are exigent due to the moderately unusual occurrence and typically short duration of packet failure episodes. The intention of our study is to understand how to determine packet failure episodes accurately with end-to-end probes. We start by taxing the ability of standard Poisson-modulated end-to-end determinations of failure in a prohibited laboratory environment using IP routers and service end hosts. Our tests show that failure characteristics reported from such Poisson-modulated probe tools can be pretty erroneous over a series of traffic conditions. Aggravated by these annotations Specifically, our method entails probe tests that follow a statistical allocation to 1) permit a precise trade-off between accuracy and force on the network, and 2) permit more accurate determinations than normal Poisson probing at the same rate. We appraise the capabilities of our slant testually by mounting and implementing a prototype tool, called VOILA. The tests express the trade-offs between force on the network and determination accuracy. We show that VOILA hearsay failure characteristics far more accurately than conventional failure determination tools.*

*Key Terms--- Dynamic Determination, VOILA, network congestion, network probes, packet failure.*

### I. INTRODUCTION

DETERMINING and analyzing network traffic dynamics between end hosts has provided the establishment for the expansion of several special network protocols and systems. Of fussy consequence understands packet failure behavior since failure can have a significant impact on the concert of both TCP- and UDP-based applications. Regardless of hard work of network engineers and operators to bound failure, it will almost certainly never be eliminated due to the essential dynamics and scaling properties of transfer in packet switched network [1]. Network operators have the knack to reflexively observe nodes within their network for packet failure on routers using SNMP. End-to-end dynamic determinations using probes provide a uniformly precious outlook since they signify the circumstances that application transfer is experiencing on those paths.

The most frequently used tools for probing end-to end paths to determine packet failure resemble the omnipresent PING function. PING-like tools drive probe packets (e.g., ICMP echo packets) to an end host at rigid intervals. Failure is indirect by the dispatcher if the response packets anticipated from the target host are not established within a particular time period. Generally speaking, a dynamic determination approach is tricky because of the distinct *sampling* environment of the probe method. Thus, the accuracy of the resulting

determinations depends both on the characteristics and interpretation of the sampling process as well as the characteristics of the underlying failure process.

Although their extensive use, there is almost no reveal in the journalism of how to refrain and regulate [2] dynamic determinations of packet failure to recover accuracy or how to best infer the ensuing determinations. One loom is recommended by the well-known PASTA principle [3] which, in a networking situation, tells us that Poisson-modulated probes will grant fair time middling determinations of a router queue's status. This proposal has been recommended as a base for dynamic determination of end-to-end interruption and failure [4]. However, the asymptotic nature of PASTA means that when it is applied in practice, the higher moments of determinations must be considered to determine the validity of the reported results. A closely related issue is the fact that failure is typically a rare event in the Internet [5]. This reality imply moreover that determinations must be taken over a protracted moment period, or that common rates of Poisson-modulated probes may have to be somewhat high in order to report accurate estimates in a timely fashion However, escalating the mean probe rate may lead to the situation that the probes themselves twist the results. Thus, there are trade-offs in packet failure determinations between probe rate, determination accuracy, impact on the path and timeliness of results.

The aim of our study is to recognize how to accurately determine failure characteristics on end-to-end paths with probes. We are engrossed in two specific features of packet failure: *failure episode frequency*, and *failure episode duration* [5]. Our study consists of three parts: (*i*) pragmatic valuation of the presently existing approach, (*ii*) expansion of assessment techniques that are based on original testual design, original probing techniques, and simple rationale tests, and (*iii*) pragmatic assessment of this new methodology.

We begin by testing standard Poisson-modulated probing in a prohibited and suspiciously instrumented laboratory environment consisting of commodity workstations separated by a series of IP routers. Background traffic is sent between end hosts at different levels of intensity to generate failure episodes thereby enabling repeatable tests over a range of conditions. We consider this setting to be ideal for testing failure determination tools since it combines the advantages of traditional simulation environments with those of tests in the wide area. Namely, much like simulation, it provides for a high level of control and an ability to compare results with "ground truth." Furthermore, much like tests in the wide area, it provides an ability to consider failure processes in actual router buffers and queues, and the behavior of *implementations* of the tools on commodity end hosts. Our tests expose two imperative deficiencies with simple Poisson probing. First, individual probes often erroneously report the dearth of a failure episode (i.e., they are successfully transferred when a failure episode is underway). Second, they are not well matched to determine failure episode duration over limited determination periods.

Our annotations about the weaknesses in standard Poisson probing inspire the second part

of our study: the advance of a new loom for end-to-end failure determination that includes four key elements. First, we intend a probe procedure that is statistically dispersed and that assesses the possibility of failure practiced by other flows that use the identical path, rather than simply reporting its personal packet deficiencies. The probe progression assumes FIFO queues along the path with a drop-tail policy. Second, we design a new testual structure with inference techniques that openly approximate the mean interval of the failure episodes without estimating the interval of any character failure episode. Our estimators are proved to be regular, under calm assumptions of the probing process. Third, we provide simple confirmation tests (that require no further trialing or data gathering) for some of the arithmetical assumptions that underlay our analysis. Finally, we confer the variance characteristics of our estimators and show that while frequency approximate difference depends only on the total number of probes emitted, failure duration variance depends on the frequency estimate as well as the number of probes sent.

The third part of our study involves the testual estimation of our new failure determination methodology. To this end, we developed a one-way dynamic determination tool called VOILA. VOILA sends fixed-size probes at specified intervals from one determination host to a collaborating target host. The end system collects the probe packets and reports the failure characteristics after a particular interval of time. We also compare VOILA with a standard tool for failure determination that emits probe packets at Poisson intervals. The results show that our tool reports failure episode estimates much more accurately for the same number of probes. We also show that VOILA estimates converge to the underlying failure episode frequency and duration characteristics.

The most significant allusion of these consequences is that there is now a methodology and tool available for wide-area studies of packet failure characteristics that enables researchers to comprehend and identify the trade-offs between accuracy and force. Additionally, the tool is self-calibrating [2] in the sense that it can report when estimates are deprived. Sensible applications could comprise its use for trail assortment in peer-to-peer overlay networks and as a tool for network operators to observe exact segments of their infrastructures.

## II. RELATED WORK

There have been many studies of packet failure behavior in the Internet. Bolot [6] and Paxson [7] evaluated end-to-end probe determinations and reported characteristics of packet failure over a selection of paths in the wide area. Yajnik *et al.* evaluated packet failure correlations on longer time scales and developed Markov models for chronological addiction structures [8]. Zhang *et al.* characterized several aspects of packet failure behavior [5]. In particular, that work reported determines of *reliability* of failure episode rate, failure episode duration, failure free period duration and overall failure rates. Papagiannaki *et al.* [9] used a sophisticated passive monitoring infrastructure inside Sprint's IP backbone to congregate packet traces and

analyze characteristics of delay and congestion. Finally, the limitations in standard end-to-end Poisson probing tools by comparing the failure rates determined by such tools to failure rates determined by passive means in a fully instrumented wide area infrastructure [10].

The foundation for the notion that Poisson Arrivals See Time Averages (PASTA) was developed by Brumelle [11], and later formalized by Wolff [3]. Variation of those queuing theory thoughts into a network probe circumstance to determine failure and interruption characteristic began with Bolot's study [6] and was extended by Paxson [7]. In recent work, Baccelli *et al.* evaluate the effectiveness of PASTA in the networking perspective [12]. Of particular significance to our work is Paxson's reference and use of Poisson-modulated dynamic probe streams to trim down preconception in delay and failure determinations. More than a few studies comprise the use of failure determinations to approximate network properties such as bottleneck buffer size and cross traffic intensity [13], [14]. The Internet Performance Determination and Analysis efforts [15], [16] resulted in a sequence of RFC's that identify how packet failure determinations should be identified. However, those RFC's are devoid of facts on how to refrain probe processes and how to deduce the consequential determinations. We are also guided by Paxson's up to date work [2] in which he advocates meticulous calibration of network determination tools.

ZING is a tool for determining end-to-end packet failure in one direction between two participating end hosts [17], [18]. ZING sends UDP packets at Poisson-modulated intervals with rigid mean rate. Savage developed the STING [19] tool to compute failure rates in both frontward and turn around directions from a single host. STING uses an intellectual scheme for manipulating a TCP stream to determine failure. Allman *et al.* confirmed how to approximate TCP failure rates from passive packet traces of TCP transfers taken close to the sender. An allied study examined in dynamic packet traces taken in the middle of the network. Network tomography based on using both multicast and unicast probes has also been verified to be helpful for inferring failure rates on inner links on end-to-end paths.

## III. DEFINITIONS OF FAILURE CHARACTERISTICS

There are many factors that can contribute to packet failure in the Internet. We illustrate some of these issues in specify as a basis for accepting our dynamic determination objectives. The background that we consider is modeled as a set of flows that exceed through a router R and contend for a single output link with bandwidth $B_{out}$ as shown in Fig. 1(a). The summative input bandwidth ($B_{in}$) must be larger than the collective output link ($B_{out}$) in order for failure to take place. The mean round trip time for the N flows is M sec. Router R is configured with Q bytes of packet buffers to put up transfer bursts, with Q naturally sized on the order of M×B [20], [21]. We presume that the queue operates in a FIFO method, that the traffic includes a concoction of short- and long-lived TCP flows as is common in today's Internet, and that the value of will ebb and flow more than time.

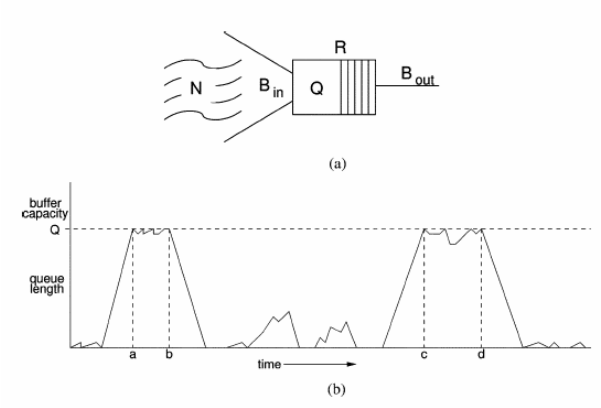


Fig 1. Simple system representation and example of failure characteristics under observation. (a) Simple system representation. N flows on input links with collective bandwidth $B_{in}$ compete for a single output link on router R with bandwidth $B_{out}$ where $B_{in} > B_{out}$. The output link has Q s of buffer capacity. (b) Example of the growth of the length of a queue over time. The queue length grows when summative demand exceeds the capacity of the output link. Failure episodes begin (points *a* and *c*) when the maximum buffer size _ is exceeded. Failure episodes end (points *b* and *d*) when aggregate demand falls below the capacity of the output link and the queue drains to zero.

Fig. 1(b) is a design of how the use of the buffer in router R may change. When the summative sending rate of the N flows exceeds the ability of the common output link, the output buffer begins to fill. This consequence is seen as a positive slope in the queue length graph. The rate of raise of the queue length depends both on the number N and on sending rate of each source. A *failure episode* begins when the collective sending rate has exceeded $B_{out}$ for a interlude of instance sufficient to load Q bytes into the output buffer of router R (e.g., at times a and c in Fig. 1(b)). A failure episode ends when the collective sending rate drops below $B_{out}$ and the buffer begins a steady drain down to zero (e.g., at times b and d in Fig. 1(b)). This typically happens when TCP sources intellect a packet failure and divide their sending rate, or simply when the number of challenging flows N drops to a sufficient level. In the former case, the duration of a failure episode is related to M, depending whether failure is sensed by a timeout or fast retransmit signal. We define *failure episode duration* as the difference between begin and end times (i.e., b-a and d-c). While this definition and model for failure episodes is somewhat basic and reliant on well behaved TCP flows, it is important for any determinant method to be stout to flows that do not react to congestion in a TCP-friendly fashion.

This definition of failure episodes can be considered a “router-centric” view since it says nothing about when any one end-to-end flow (including a probe stream) actually loses a packet or wits a lost packet. This contrasts with most of the prior work discussed in Section II which consider only failures of individual or groups of probe packets. In other words, in our methodology, a failure episode begins when the probability of some packet failure becomes positive. During the episode, there might be transient periods during which packet failure ceases to occur, followed by resumption of some packet failure. The episode ends when the probability of packet failure stays at 0 for a sufficient period of time (longer than a typical RTT). Thus, we offer two definitions for *packet failure rate*:

• **Router-centric failure rate**: With L the number of dropped or failure packets on a given output link on router R during a given period of time, and S the number of all effectively transmitted packets through the same link over the same period of time, we define the router-centric failure rate as *L/(S+L).*

• **End-to-end failure rate**: We define end-to-end failure rate in accurately the same manner as router-centric failure-rate, with the warning that we

only add up packets that belong to a detailed flow of interest.

It is important to differentiate between these two notions of failure rate since packets are transmitted at the utmost rate during failure episodes. The result is that during a period where the router-centric failure rate is non-zero, there may be flows that do not fail any packets and therefore have end-to-end failure rates of zero. This surveillance is central to our study and bears directly on the design and implementation of active determinant methods for packet failure.

As a corollary, an important thought of our probe process described below is that it must deal with instances where entity probes do not accurately report failure. We therefore differentiate between the *true failure episode state* and the *probe-determined* or *observed state*. The former refers to the router-centric or end-to-end jamming state, given cherished knowledge of buffer occupancy, queuing delays, and packet drops, e.g., in order hidden in the queue length graph in Fig. 1(b). Ideally, the probe-determined state reflects the true state of the network.

## IV. Estimation Of Plain Poisson Probing For Packet Failure

We begin by using our laboratory to estimate the capabilities of simple Poisson-modulated failure probe determination using the ZING tool [17], [18]. ZING determines packet waiting and failure in one direction on an end-to-end path. The ZING sender sends UDP probe packets at Poisson-modulated mean times with timestamps and unique serial numbers and the receiver logs the probe packets incoming. Users identify the mean probe rate, the probe packet size, and the number of packets in a "fledge."

To estimate simple Poisson probing, we configured ZING using the same parameters as in [5]. Specifically, we trotted two tests, one with $\lambda$=200ms (10 Hz) and 512 byte payloads and another with $\lambda$=100ms (20 Hz) and 128 byte payloads. To determine the extent of our tests below, we preferred an epoch of time that should bound the difference of the failure rate estimator $\bar{X}$ where Var($\bar{X}_n$) $\approx p/n$ for failure rate $p$ and number of probes $n$.

We conducted three different tests in our estimation of simple Poisson probing. In each test we determined both the frequency and duration of packet failure episodes. Again, we used the definition in [5] for failure episode: "a sequence of successive packets (perhaps only of same length) that were lost."

The first test used 40 infinite TCP sources with receive windows set to 512 full size (1500 bytes) packets. Fig. 2(a) shows the instance sequence of the queue possession for a part of the test; the predictable management actions of TCP sources in jamming evasion is apparent. The test was run for a period of 15 min which should have enabled ZING to determine failure rate with standard deviation within 10% of the mean [10].

Results from the test with infinite TCP sources are shown in Table I. The table shows that ZING performs badly in determining both failure frequency and period in this situation. For both probe rates, there were no instances of repeated failure packets, which clarify the lack of ability to evaluate failure episode period.

In the second set of tests, we used Viden to generate a sequence of (approximately) steady period (about 68 ms) failure episodes that were spaced arbitrarily at exponential intervals with mean of 10s over a 15 minute period. The time sequence of the queue length for a portion of the test period is shown in Fig. 2(b).

Results from the test with arbitrarily spaced, constant duration failure episodes are shown in Table II. The table shows that ZING determines failure frequencies and durations that are nearer to the true values.

In the ultimate set of tests, we used Harpoon to generate a sequence of failure episodes that approximate failure resultant from web-like traffic. Harpoon was configured to briefly raise its load in order to bring packet failure, on average, every 20s. The inconsistency of traffic produced by Harpoon complicates description of failure episodes. To start baseline failure episodes to evaluate against, we found trace segments where the primary and very last events were packet failures, and queuing delays of all packets between those failures were above 90 ms (within 10 ms of the maximum).We trotted this test for 15 min and a portion of the time series for the queue length is shown in Fig. 2(c).

Results from the test with Harpoon web-like traffic are shown in Table III. For determining failure frequency, neither probe rate results in a near match to the correct frequency. For failure episode duration, the results are also deprived. For the 10 Hz probe rate, there were no successive failures determined, and for the 20 Hz probe rate, there were only two instances of successive failures, each of closely two lost packets.

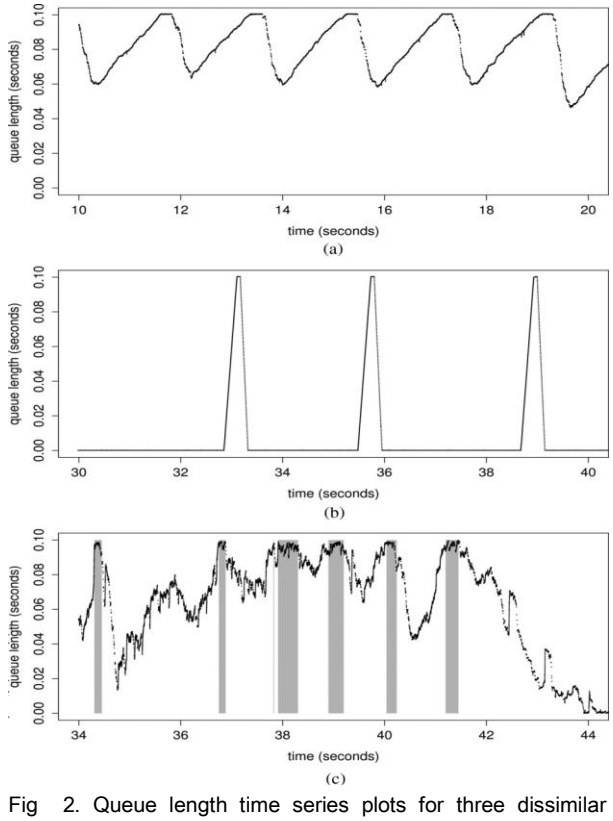


Fig 2. Queue length time series plots for three dissimilar background traffic scenarios. (a) Queue length time series for a portion of the experiment with 40 infinite TCP sources. (b) Queue length time series for a portion of the experiment with randomly spaced, constant duration failure episodes. (c) Queue length time series for a portion of the experiment with Harpoon web-like traffic. Time segments in gray indicate failure episodes.

**TABLE I**

RESULTS FROM ZING EXPERIMENTS WITH INFINITE TCP SOURCES

| | Frequency | Duration Mean(standard deviation)(seconds) |
|---|---|---|
| True Values | 0.0268 | 0.138 (0.0091) |
| ZING (10 HZ) | 0.0007 | 0 (0) |
| ZING (20 HZ) | 0.0003 | 0 (0) |

**TABLE II**

RESULTS FROM ZING EXPERIMENTS WITH ARBITRARILY SPACED, STABLE DURATION FAILURE EPISODES

| | Frequency | Duration Mean(standard deviation)(seconds) |
|---|---|---|
| True Values | 0.0072 | 0.070 (0.000) |
| ZING (10 HZ) | 0.0039 | 0.045 (0.0011) |
| ZING (20 HZ) | 0.0034 | 0.053 (0.0023) |

**TABLE III**

RESULTS FROM ZING EXPERIMENTS WITH HARPOON WEB-LIKE TRAFFIC

| | Frequency | Duration Mean(standard deviation)(seconds) |
|---|---|---|
| True Values | 0.0096 | 0.139 (0.0089) |
| ZING (10 HZ) | 0.0017 | 0 (0) |
| ZING (20 HZ) | 0.0015 | 0.025 (0.0011) |

## V. PROBE TOOL PERFORMANCE AND ESTIMATION

To estimate the capabilities of our failure probe determination process, we built a tool called VOILA[1] that trapping the basic algorithm. We then conducted a sequence of experiments with VOILA in our laboratory with the same background traffic scenarios described in Section IV.

The goal of our lab-based tests was to confirm our modeling process and to estimate the capability of VOILA over a variety of failure situations. We report results of experiments focused in three areas. At the same time as our probe procedure doesn't guess that we constantly get true indications of failure from our probes, the accuracy of reported determination will develop if probes more constantly point to failure. Among this in mind, the first set of tests was considered to recognize the capability of an individual probe (consisting of 1 to N tightly-spaced packets) to accurately report an encounter with a failure episode. The second is to check the accuracy of VOILA in reporting failure episode frequency and period for a range of probe rates and traffic scenarios. In our final set of tests, we match up to the capabilities of VOILA with simple Poisson-modulated probing.

### *A. Exact Reporting of Failure Episodes by Probes*

We distinguished in Section III that, preferably, a probe should give an exact suggestion of the true failure episode state [(1)]. Though, this may not be the case. The main matter is that during a failure episode, many packets go on to be successfully transmitted. Thus, we hypothesized that we might be capable to increase the chances of probes correctly reporting a failure episode by increasing the amount of packets in an individual probe. We also hypothesized that, assuming FIFO queuing, using one-way holdup information could additional develop the accurateness of individual probe determination.

We investigated the first hypothesis in a sequence of tests using the infinite TCP source background traffic and constant-bit rate traffic described in Section IV. Intended of the infinite TCP traffic, failure event durations were roughly 150ms. For the constant-bit rate traffic, failure episodes were roughly 68 ms in duration. We used a customized edition of VOILA to produce probes at permanent intervals of 10 ms so that some number of probes would bump into all failure episodes. We tested with probes consisting of between 1 and 10 packets. Packets in an individual probe were sent back to back per the capabilities of the determinant hosts (i.e., with approximately 30$\mu s$ between packets). Probe packet sizes were set at 600 bytes.

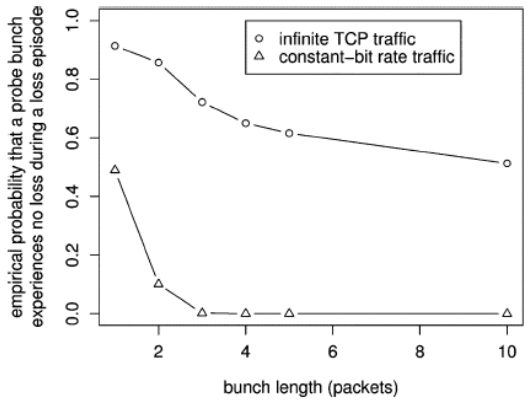


Fig. 3. Results from tests of ability of probes containing of N packets to report failure when an episode is encountered.

*B. Determining Frequency and Duration*

The formulation of our new failure probe procedure calls for the user to identify two parameters, and, where is the possibility of initiating a essential test at a given interval. In the next set of experiments, we explore the effectiveness of VOILA to report failure episode frequency and period for a permanent *N* and *p* using values of 0.1, 0.3, 0.5, 0.7, and 0.9 (implying that probe traffic frenzied between 0.2% and 1.7% of the restricted access link). With the time discretization set at 5 ms, we set for these experiments at 240,000 squashy test duration of 900s. We also test the failure frequency and duration estimates for a fixed *p* of 0.1 and *N* of 960,000 from an hour-long experiment.

In these tests, we used three dissimilar background traffic scenarios. In the first scenario, we used Viden to produce accidental failure episodes at static duration as described in Section IV. For the second, we customized Viden to generate failure episodes of three different durations (50, 100, and 150 ms), with an average of 10s between failure episodes. In the final traffic scenario, we used Harpoon to generate self-similar, web-like workloads as described in Section IV. For all traffic scenarios, VOILA was configured with probe sizes of 3 packets and with packet sizes fixed at 600 bytes. The three packets of each probe were sent back-to-back, according to the abilities of our end hosts (approximately 30 *μs* between packets). For each probe rate, we set $\tau$ to the expected time between probes plus one standard deviation (viz., $\tau = (1-p/p) + \sqrt{(1-p/p^2)}$ time slots). For α, we used 0.2 for probe probability 0.1, 0.1 for probe probabilities of 0.3 and 0.5, and 0.05 for probe probabilities of 0.7 and 0.9.

**TABLE III**

COMPARISION OF FAILURE ESTIMATES FOR *p=0.1* AND TWO DIFFERENT VALUES OF *N* AND TWO DIFFERENT VALUES FOR THE $\tau$ THRESHOLD PARAMETER

| N | $\tau$ | Failure Frequency | | Failure Duration (seconds) | |
|---|---|---|---|---|---|
| | | True | VOILA | True | VOILA |
| 240,000 | 40 | 0.0064 | 0.0008 | 0.072 | 0.025 |
| 240,000 | 80 | 0.0064 | 0.0021 | 0.072 | 0.057 |
| 960,000 | 40 | 0.0064 | 0.0011 | 0.072 | 0.023 |
| 960,000 | 80 | 0.0064 | 0.0027 | 0.072 | 0.046 |

*C. Active Features of the Evaluators*

As we have shown, estimates for a small probe rate do not considerably pick up even with rather large *N*. A reserved enlarge in the probe rate *p*, however, significantly improves the correctness and convergence time of both frequency and period estimates. Fig. 4 shows results from an experiment using Harpoon to create self-similar, web-like TCP traffic for the failure episodes. For this test, *p* is set to 0.5. The peak plot shows both

the dynamic features of both true and estimated failure episode frequency for the complete 15 min-long experiment. VOILA estimates are produced every 60s for this test. The error bars at each VOILA estimate indicate a 95% buoyancy interval for the estimates. We see that even after 1 or 2 min, VOILA estimates have converged close to the true values. We also see that VOILA tracks the true frequency reasonably well. The bottom plot in Fig. 4 compares the true and estimated characteristics of failure episode duration for the same test. Again, we see that after a short period, VOILA estimates and poise intervals have converged close to the true mean failure episode duration. We also see that the active behavior is generally well followed. Except for the low probe rate of 0.1, results for other experiments exhibit similar qualities.

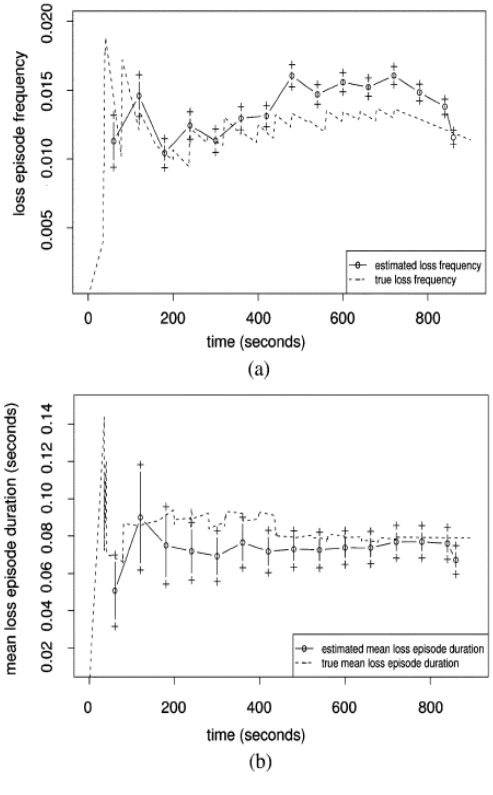


Fig 4. Comparison of failure frequency and duration estimates with true values over 15 min for Harpoon web-like cross traffic and a probe rate $p = 0.5$. VOILA estimates are produced every minute, and error bars at each estimate indicate the 95% confidence interval. Top plot shows results for failure episode frequency and bottom plot shows results for failure episode duration.

## VI. USING VOILA IN PRACTICE

There are a number of important realistic issues which must be considered when using VOILA in the wide area:

- The tool requires the user to select values for $p$ and $N$. Assume for now that the number of failure events is immobile over time. Let $B_0$ be the mean number of failure events that occur over a unit period of time. For example, if a regular of 12 failure events occur every minute, and our discretization unit is 5 ms, then $B_0$ = 12/(60×200) = 0.001 (this is, of course, an estimate of the true the value of $B_0$). With the inactive assumption on $B_0$, we imagine the accuracy of our evaluators to depend on the product $pN$ $B_0$, but not on the individual values of $p$, $N$ or $B_0$. Indeed, that a dependable rough calculation of the relative standard deviation in our estimation of duration is given by

  Standard Deviation (duration) $\approx \frac{1}{\sqrt{2\,pNB_0}}$

- The recent study on packet failure via passive determinant [9] indicates that failure episodes in backbone links can be very short-lived (e.g., on the order of several microseconds). The only condition

for our tool to effectively detect and approximate such short durations is for our discretization of time to be finer than the order of time period we attempt to approximate. Such an obligation may imply that commodity workstations cannot be used for precise active determinant of end-to-end failure characteristics in some conditions. A consequence to this is that active determinants for failure in high bandwidth networks may need high-performance, specialized systems that support small time discretization.

- Our assessment of duration is seriously based on accurate evaluation of the ratio *B/M*. We approximate this ratio by counting the occurrence rate of *Y*=01, as well as the occurrence rate of *Y*=10. The number *B/M* can be expected as the average of these two rates. The *justification* is done by determining the *differentiation* between these two rates. This variation is directly relative to the expected standard deviation of the above estimation.
- Our categorization of whether a probe traversed a congested path concerns not only whether the probe was failed, but how long it was deferred. While a suitable $\tau$ parameter appears to be dictated primarily by the value of *p*, it is not yet clear how best to set α for an arbitrary path, when characteristics such as the level of statistical multiplexing or the physical path configuration are unknown. Examination of the sensitivity of $\tau$ and α in more intricate environment is a subject for future work.
- To accurately calculate end-to-end wait time for inferring congest ion requires time synchronization of end hosts. While we can trivially abolish offset, clock tilt is still a unease. New on-line synchronization techniques such as reported in [22], or even off line methods such as [23] could be used effectively to address this matter.

## VII. Summary, Conclusions And Future Work

The intention of our study was to recognize how to determine end-to-end packet failure individuality accurately with probes and in a way that enables us to specify the impact on the bottleneck queue. We began by estimating the abilities of simple Poisson-modulated probing in a controlled laboratory environment consisting of commodity end hosts and IP routers. We think about this for failure determination tool evaluation since it activates repeatability, concern of ground truth, and a range of traffic conditions under which to subject the tool. Our primary tests point out that simple Poisson probing is relatively ineffective at measuring failure episode frequency or determining failure episode duration, particularly when subjected to TCP (immediate) cross traffic.

These trial results lead to our improvement of a statistically distributed probe procedure that provides more exact evaluation of failure characteristics than simple Poisson probing. The testual design is constructed in such a way that the recital of the associated evaluators relies on

the total number of probes that are sent, but not on their sending rate. Moreover, simple techniques that agree to user to authenticate the measurement output are introduced. We executed this method in a new tool, VOILA, which we tested in our laboratory. Our tests exhibits that VOILA, in most cases, accurately evaluate loss frequencies and durations over a range of cross traffic conditions. For the same overall packet rate, our results show that VOILA is considerably more exact than Poisson probing for determining failure episode characteristics.

While VOILA enables superior accuracy and a better accepting of link impact versus timeliness of determination, there is still room for improvement. First, we intend to examine why $p=0.1$ does not appear to work well even as $N$ increases. Second, we plan to examine the issue of suitable parameterization of VOILA, as well as packet sizes and the α and $\tau$ parameters, over a range of reasonable operational settings with more multifarious multihop paths. At last, we have considered adding adaptively to our probe process model in a limited sense. We are also considering alternative, parametric methods for inferring failure characteristics from our probe process. Another task is to approximate the inconsistency of the estimates of congestion frequency and time period themselves straightly from the determined data, under a minimal set of geometrical assumptions on the congestion procedure.

## ACKNOWLEDGEMENT

We, authors express gratitude to all the anonymous reviewers for their affirmative annotations among our paper.

## REFERENCES

[1] W. Leland, M. Taqqu, W. Willinger, and D. Wilson, "On the self- similar nature of Ethernet traffic (extended version)," *IEEE/ACM Trans.Networking*, vol. 2, no. 1, pp. 1–15, Feb. 1994.

[2] V. Paxson, "Strategies for sound internet measurement," in *Proc. ACM SIGCOMM '04*, Taormina, Italy, Nov. 2004.

[3] R. Wolff, "Poisson arrivals see time averages," *Oper. Res.*, vol. 30, no 2, Mar.–Apr. 1982.

[4] G. Almes, S. Kalidindi, and M. Zekauskas, "A one way packet loss metric for IPPM," IETF RFC 2680, Sep. 1999.

[5] Y. Zhang, N. Duffield, V. Paxson, and S. Shenker, "On the constancy of internet path properties," in *Proc. ACM SIGCOMM Internet Measurement Workshop '01*, San Francisco, CA, Nov. 2001.

[6] J. Bolot, "End-to-end packet delay and loss behavior in the internet," in *Proc. ACM SIGCOMM '93*, San Francisco, CA, Sep. 1993.

[7] V. Paxson, "End-to-end internet packet dynamics," in *Proc. ACM SIGCOMM '97*, Cannes, France, Sep. 1997.

[8] M. Yajnik, S. Moon, J. Kurose, and D. Towsley, "Measurement and modeling of temporal dependence in packet loss," in *Proc. IEEE INFOCOM '99*, New York, Mar. 1999.

[9] D. Papagiannaki, R. Cruz, and C. Diot, "Network performance monitoring at small time scales," in *Proc. ACM SIGCOMM '03*, Miami, FL, Oct. 2003.

[10] P. Barford and J. Sommers, "Comparing probe- and router-based packet loss measurements," *IEEE Internet Computing*, Sep./Oct. 2004.

[11]S. Brumelle, "On the relationship between customer and time averages in queues," *J. Appl. Probabil.*, vol. 8, 1971.

[12]F. Baccelli, S. Machiraju, D. Veitch, and J. Bolot, "The role of PASTA in network measurement," in *Proc. ACM SIGCOMM*, Pisa, Italy, Sep. 2006.

[13]S. Alouf, P. Nain, and D. Towsley, "Inferring network characteristics via moment-based estimators," in *Proc. IEEE INFOCOM '01*, Anchorage, AK, Apr. 2001.

[14]K. Salamatian, B. Baynat, and T. Bugnazet, "Cross traffic estimation by loss process analysis," in *Proc. ITC Specialist Seminar on Internet Traffic Engineering and Traffic Management*, Wurzburg, Germany, Jul. 2003.

[15]Merit Internet Performance Measurement and Analysis Project. 1998 [Online]. Available: http://www.nic.merit.edu/ipma/

[16]Internet Protocol Performance Metrics. 1998 [Online]. Available: http://www.advanced.org/IPPM/index.html

[17]A. Adams, J. Mahdavi, M. Mathis, and V. Paxson, "Creating a scalable architecture for Internet measurement," *IEEE Network*, 1998.

[18]J. Mahdavi, V. Paxson, A. Adams, and M. Mathis, "Creating a scalable architecture for Internet measurement," in *Proc. INET '98*, Geneva, Switzerland, Jul. 1998.

[19]S. Savage, "Sting: A tool for measuring one way packet loss," in *Proc. IEEE INFOCOM '00*, Tel Aviv, Israel, Apr. 2000.

[20]G. Appenzeller, I. Keslassy, and N. McKeown, "Sizing router buffers," in *Proc. ACM SIGCOMM*, Portland, OR, 2004.

[21]C. Villamizar and C. Song, "High performance TCP in ASNET," *ACM Comput. Commun. Rev.*, vol. 25, no. 4, Dec. 1994.

[22]A. Pasztor and D. Veitch, "PC based precision timing without GPS," in *Proc. ACM SIGMETRICS*, Marina Del Ray, CA, Jun. 2002.

[23] L. Zhang, Z. Liu, and C. Xia, "Clock synchronization algorithms for network measurements," in *Proc. IEEE INFOCOM*, New York, Jun. 2002.