This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCS.2020.2978231, IEEE Open Journal of the Computer Society

A Simulation Study of the Measurement of Queueing Delay over End-to-End Paths

Khondaker Salehin, *Senior Member, IEEE*, Ki Won Kwon, and Roberto Rojas-Cessa, *Senior Member, IEEE*

Determining the qualitative states of the Internet requires an accurate knowledge of queueing delay over an end-to-end path. However, the measurement of queueing delay in a large network is still considered a complex and open problem. Existing schemes that measure queueing delay compensate for this complexity using a high infrastructural support and administrative access to the path under test even though their feasibility and accuracy on the Internet are low. In this paper, we propose an active scheme, called COMPRESS: COMpound Probe compRESSion, to measure queueing delay on all routers over an end-to-end path. The proposed scheme performs per-hop measurement using UDP-based probing packets. It is both simple and self-sufficient in comparison to the existing schemes. We have implemented the proposed scheme in a simulation environment to present a controlled performance evaluation under different levels (e.g., light, moderate, and heavy) and types (e.g., symmetric and asymmetric) of queueing delays over single- and multiple-hop paths. Our simulation results show that the scheme is sensitive to the induced queueing delays and consistently provides a high measurement accuracy. Overall, the scheme has an average measurement error of around 20% or below over the simulated paths.

***Index Terms*—Measurement techniques, computer simulation, packet-pair model, queuing delay, quality of service, end-to-end path.**

I. Introduction

END-to-end delay is a critical parameter of the Internet infrastructure. It is defined by both static and dynamic aspects of its constituent parameters. The static aspect consists of three parameters: i) packet processing time (e.g., packet scheduling and switching latencies) of Internet routers and workstations [1], [2], ii) packet transmission time over network links, and iii) propagation time of network links over an end-to-end path. The dynamic aspect consists of only one parameter called queueing delay. This parameter refers to the additional waiting time inside routers that a packet experiences due to the competing traffic flows over the end-to-end path at any instance of time [1], [3], [4], [5], [6].

For example, Figure 1 shows two separate scenarios that illustrate queueing delay on a router, node i, consisting of multiple input links (L_1, ..., L_n) and one output link (L_o). This figure represents the first-in-first-out queueing mechanism, which is widely used in the Internet devices [33]. In Figure 1(a), node i carries only one traffic flow from L_n to L_o; therefore, the traffic packets (P_1, P_2, and P_3) do not experience any additional waiting time in the output queue (Queue o) before their transmissions through L_o. At a different instance of time, as Figure 1(b) shows, the same node i carries two different traffic flows from L_1 and L_n, respectively, to L_o as Queue o holds a packet C_1 from L_1 ahead of the packet P_1 from L_n. Here, P_1 experiences a queueing delay (i.e., an additional waiting time), equivalent to the transmission time of C_1 over L_o, before its own transmission through the output link in contrast to Figure 1(a).

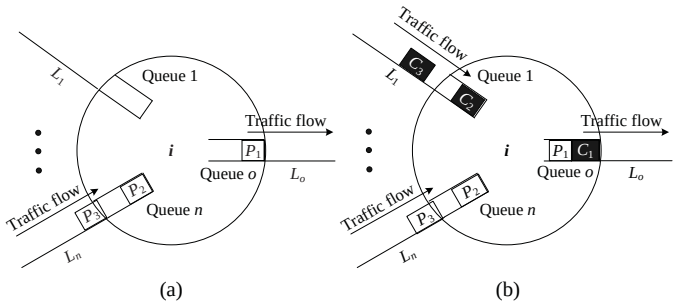


Fig. 1: A router, node i, carrying (a) one traffic flow and (b) two traffic flows at two different instances of time.

With the continuing technological advancements of Internet infrastructure, there is an inclination to assume queueing delay as an insignificant network parameter for determining the qualitative states of the Internet [5]. However, such an assumption is inaccurate because of the availability of non-trivial (i.e., non-zero) queueing delays over Internet paths [5], [7]. This observation suggests that the measurement of queueing delay on the Internet routers is important. Moreover, it is essential for successful deployments of various delay-sensitive applications that demand a strict quality of service and service-level agreements.

For example, accurate one-way-delay (OWD) measurement with a precise knowledge of queueing delay over an end-to-end path is necessary for achieving a high accuracy in triangulation-based IP geolocation [8], [9], [3]. The same knowledge is useful for improving the reliability of electronic-trading applications for an enhanced customer confidence [2], [10]. Applications in Tactile Internet require a stringent budget on end-to-end delays [11]; therefore, a precise measurement of

The manuscript was submitted for review on Jan 22, 2020. This work is partially supported by a start-up grant from the College of Natural & Behavioral Sciences, California State University, Dominguez Hills.

K. Salehin is with the Department of Computer Science, California State University, Dominguez Hills, Carson, CA 90747, USA (e-mail: ksalehin@csudh.edu).

K. Kwon was with the Computer Science Program, Bard College, Annandale-on-Husdon, NY 12504, USA (e-mail: billykwon111@gmail.com).

R. Rojas-Cessa is with the Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102, USA (e-mail: rojas@njit.edu).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCS.2020.2978231, IEEE Open Journal of the Computer Society

queueing delay is essential in this field as well. Nevertheless, the measurement of this parameter on a large network remains a complex and open problem. The complexity here stems from the dynamic nature of the parameter itself [5]. In addition, we need a close co-operation (i.e., infrastructural support and administrative access) from the Internet for a successful measurement of queueing delay according to the current state-of-the-art [4].

In this paper, we propose a simple scheme to actively measure queueing delay. The proposed scheme identifies changes (i.e., compressions) in pairs of its probing packets to estimate queueing delays on all routers over an end-to-end path. The scheme is self-sufficient because it uses User Datagram Protocol (UDP) packets and performs the measurement without requiring a close co-operation from the path. A simulation study using different traffic conditions shows the scheme is efficient and accurate over both single- and multiple-hop paths.

We organize the rest of the paper as follow: Section II presents the state-of-the-art of queueing-delay measurement. Section III describes the basic principle and the proposed scheme in detail. Section IV presents the simulation study of the proposed scheme. Section V outlines the directions for future research. Section VI concludes the discussion.

II. Related Work

Two broad categories divide existing schemes for measuring queueing delay: i) passive methods and ii) active methods. Passive methods use ongoing data traffic over a path to perform measurement whereas active methods use self-induced probing packets on the path under test to do the same. We briefly discuss existing schemes from these two categories below.

A. Passive Methods

Instrumentation is a widely-used method for measuring queueing delay on a router with direct access to the networks. A prominent study on the Sprint-backbone network measured queueing delay on its core routers by instrumenting their input and output links with specialized packet-capture equipment [12]. A similar study extended this work on commodity workstations operating as virtualized routers using software-based instrumentation at their input and output links [13]. It is evident that these studies are not suitable for remote measurement over an end-to-end path because Internet only provides restrictive physical and administrative access due to its proprietary infrastructure.

Analysis of Internet traffic is another known method for passive measurement of queueing delay. One study captured a large-scale TCP traffic and fed it to a stochastic model for determining distributions of queueing delay on the router under test [14]. But the efficacy of this work is strictly tied to the prior knowledge of traffic intensity, flow-size distribution, and packet-loss probability of the captured data set. Another study measured OWDs of the data packets collected from an ingress router to an egress router across an operational tier-1 network to identify various contributing factors, e.g., queueing delay, that add to the network delay performance [15]. Because this work relies on the captured data set inside a controlled network environment, its application is not expandable to a large-scale measurement on the Internet.

Architecture-specific measurement is popular for determining queueing delay in specialized use cases. One study leveraged a delay-based congestion control mechanism to measure large queueing delays, also known as bufferbloat [16], on BitTorrent networks [17]. Another work utilized the control plane of software defined network (SDN) to estimate queueing delay by modeling its traffic flows [18]. However, the application of these methods is limited because their implementations are specific to the underlying specifications, protocol, or network architecture.

B. Active Methods

Pathchar measures queueing delay using round trip times (RTTs) of Internet Control Message Protocol (ICMP) packets [19]. The basic idea of this scheme is that the difference between the minimum and average RTTs identifies queueing delay over an end-to-end path when the minimum RTT corresponds to a zero queueing delay. It iteratively sends trains of ICMP packets with different sizes to measure the minimum and average RTTs as it determines queueing delay on each router along the path in a hop-by-hop manner. Because pathchar uses linear-regression model for processing of its data samples, it requires a long measurement time and is considered network intrusive (i.e., the generated ICMP packets affect the traffic flows on the path under test) [20].

Schemes based on statistical inference measure queueing delay on a router using unicast and multicast packets by applying a network-tomography concept [21], [22]. Here, pairs of source and destination nodes exchange probing packets to estimate OWDs of the constituent end-to-end paths that share common network links. Packets usually experience similar OWDs on a shared link; therefore, these schemes infer queueing delays on routes over the shared links using a complex statistical processing of the collected OWD samples. The challenge with these schemes is the requirement for a strict topological configuration of the end-to-end paths inside an unregulated network.

Another popular scheme, called cing, measures queueing delay using pairs of probing packets where the separation (i.e., intra-probe gap) between the last bits of the packets in each pair provides an estimate of the parameter on the router under test [23]. It sends two ICMP echo-request packets without any separation (i.e., back-to-back) toward two neighbor routers over an end-to-end path. Here, the difference between the timestamps generated by neighbor routers for the respective ICMP echo-reply packets determines the intra-probe gap, which contains information about the queueing delay on the first of the two neighbor routers. cing sends multiple pairs of probes to estimate the smallest and average intra-probe gaps from a large data set; then estimates the queueing delay from the deference between these two statistical aggregates. This scheme repeats the above process on all routers along the path in a hop-by-hop manner. It is the simplest among all existing active methods, to the best of our knowledge; however, its dependency on ICMP packets is an impediment to its feasibility and accuracy on the Internet [23].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCS.2020.2978231, IEEE Open Journal of the Computer Society

In brief, passive methods are used for specialized measurements, e.g., single-hop and bufferbloat, whereas active methods are used for generalized measurements, and both of them lack in deployability and efficacy. Therefore, accurately characterizing queueing delay with a minimal or no cooperation from the Internet remains an open problem as of today.

III. Proposed Scheme

We propose an active scheme to measure queueing delay on routers using a packet-pair structure, known as compound probe [24]. We named the scheme as COMPRESS: COMpound Probe compRESSion because it utilizes the change in its probing structure on each router for performing the measurement. It is a simple scheme because it uses UDP packets in the probing structure. It is also self-sufficient because it does not require any infrastructural support and administrative access to the end-to-end path under test.

The basic principle of this scheme is when a compound probe arrives at a router with a pre-defined intra-probe gap, a compression in the probing structure due to cross traffic corresponds to a queueing delay [4]. Figure 2 illustrates an example of a queueing delay on node i from the compression in the intra-probe gap of a compound probe, which consists of a small heading packet (P_h) followed by a large trailing packet (P_t). In Figure 2(a), P_h and P_t arrive at node i back-to-back and experience a dispersion (δ_i) in the intra-probe gap (G_i) due to the difference between their transmission times on the output (L_{i+1}) and input (L_i) links, respectively. On the other hand, Figure 2(b) has a smaller G_i on L_{i+1} than that in 2(a) since the compound probe is interrupted by a cross-traffic packet, C, at node i. We provide a magnified view of node i in Figure 2(c). This figure shows how C interferes P_h in the output queue (Queue $i+1$) that delays the transmission of P_h producing a compressed (i.e., smaller) G_i on L_{i+1}. Figure 2 suggests that the queueing delay on node i can be measured if the compression in G_i is identified with and without the presence of cross-traffic interference. This measurement principle is applicable to a single-hop scenario where node i is the only router over an end-to-end path.

We have extended the single-hop concept to a multiple-hop scenario where node i is a router over a n-hop end-to-end path such that $1 \leq i \leq n-1$, as Figure 3 shows. This extension incorporates a mechanism for avoiding the added measurement challenge at node i contributed by the cross-traffic interference and dispersion in the compound probe on all routers (node 1 to node $i-1$) of the end-to-end path. The mechanism is summarized in Figure 4, where the compound probe is equipped with a large redundant packet (P_r) ahead of both P_h and P_t. P_r is limited by a time-to-live (TTL) value, i.e., TTL = i; it ensures the probing structure experiences a zero-dispersion gap up to node $i-1$ when node i is of measurement interest. Figure 5 illustrates queueing-delay measurement at node i in detail. Because P_r is discarded at the router under test, the compound probe creates a pre-defined dispersion between P_h and P_h so the compression in the intra-probe gap on L_{i+1} contains the information about queueing delay at node i in reference to the aforementioned basic principle.

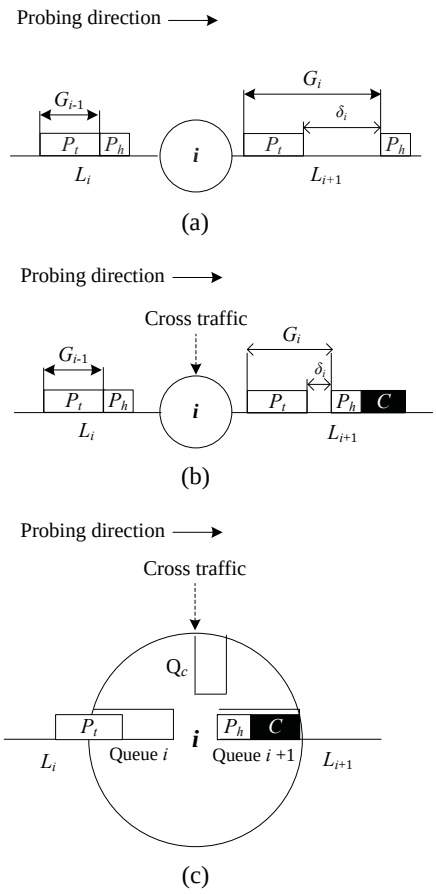


Fig. 2: A compound probe with a resultant intra-probe gap on L_{i+1} of node i (a) without and (b) with cross-traffic interference, and (c) a magnified view of node i showing the effect of a cross-traffic packet C ahead of P_h at Queue $i+1$ that creates the compression.

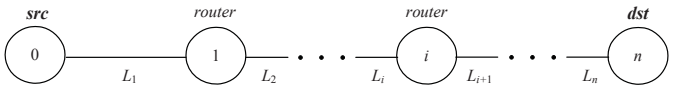


Fig. 3: A multiple-hop path between src (node 0) and dst (node n) that consists of n links and $n-1$ routers, where $n > 2$.

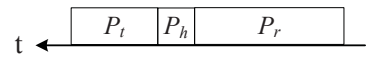


Fig. 4: A compound probe with heading packet (P_h), trailing packet (P_t), and a large redundant packet (P_r) in order.

On a multiple-hop path, maintaining a zero-dispersion gap before creating a per-defined dispersion gap in the compound probe at the router under test is feasible with the proper sizing of P_t, P_h, and P_r. However, heterogenous link-capacities over an end-to-end path can impact the sizing of the compound probe [25]. For example, any dispersion (zero and non-zero)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCS.2020.2978231, IEEE Open Journal of the Computer Society

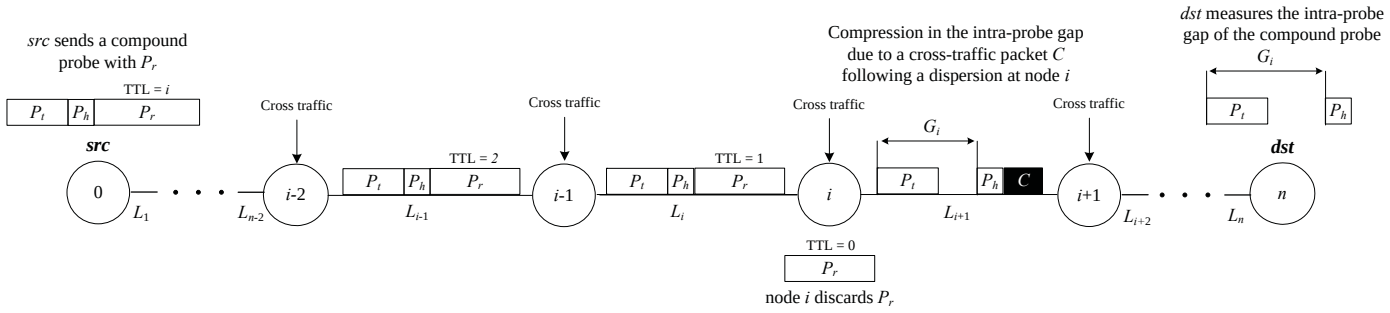


Fig. 5: Compound probe experiences a compression due to a cross-traffic packet C at node i after P_r is discarded so that the measured gap G_i contains a cumulative queueing delay from node i to node $n-1$ over a multiple-hop path.

in a pair of packets on a router is dependent on the packet-size ratio and the link-capacity ratio between the input and output links [26], [27], [25]. This infers that we need an a-priori knowledge of the link capacities over an end-to-end path for determining the required sizing of the packets in the compound probe. We do not consider this constraint as an impediment toward the self-sufficiency of our scheme. For example, existing schemes for bandwidth measurement are capable of readily estimating the link capacities over an end-to-end path with a high accuracy [19], [20], [24], [26], [27], [28].

We present the detailed steps for iteratively measuring queueing delay on every router over a multiple-hop path below. In the following steps, we refer to the n-hop path between node 0 (src) and node n (dst) in Figure 3 and consider s_r, s_h, and s_t as the sizes for P_r, P_h, and P_t, respectively.

1. Estimate link capacities, l_1, ..., l_n, over the path using a bandwidth-measurement tool.
2. Identify the smallest and the largest link-capacity ratios, lr_s and lr_l, respectively, from the estimated link capacities.
3. Consider node i, where $i = n-1$.
4. Determine s_h and s_t in reference to a packet-size ratio between s_h and s_t, $\alpha = \frac{s_h}{s_t}$ such that

$$\alpha < lr_s \tag{1}$$

5. Calculate the expected dispersion at node i between P_h and P_t as

$$\delta_i = \left(\frac{s_t}{i_i} - \frac{s_h}{l_{i+1}} \right) \tag{2}$$

This constraint in (2) refers to the maximum queueing delay that can be measured at node i.

6. Calculate the expected intra-probe gap between P_h and P_t at dst as

$$E[G_i] = \frac{s_t}{l_n} + \sum_{i}^{n-1} \delta_i \tag{3}$$

This gap refers to the zero-queueing delay due to no cross-traffic on node i.

7. Determine a large s_r considering a packet-size ratio between s_r and $s_h + s_t$, $\beta = \frac{s_r}{s_h + s_t}$, such that

$$\beta \geq lr_l \tag{4}$$

If the required s_r is larger than the maximum transmission unit (MTU) over the path, use multiple P_rs such that the cumulative size $x.s_r$, where x is an integer and s_r = MTU, satisfies (4). The constraint in (4) identifies the required s_r for maintaining a zero-dispersion gap in the compound probe up to node i.

8. Send a compound probe, headed by a TTL-limited P_r, from src to dst using the packet sizes determined in Steps 4 & 7 and TTL = i so that the compound probe discards P_r at node i.
9. Generate timestamps for P_h and P_t upon receiving them at dst to measure the intra-probe gap of the compound probe, i.e.,

$$M[G_i] = \frac{s_t}{l_n} + \sum_{i}^{n-1} \delta_i{}', \tag{5}$$

where $\delta_i{}'$ is the cumulative dispersion from node i in reference to the queueing delay, w_i, i.e.,

$$\delta_i' = \begin{cases} \delta_i - w_i & if\ \delta_i - w_i > 0 \\ 0 & else \end{cases}$$

10. Determine queueing delay at node i from the compression in the compound probe as

$$w_i = (E[G_i] - M[G_i]) - \sum_{i+1}^{n-1} w_i, \tag{6}$$

where $w_i \geq 0$.

11. Update i as $i = i - 1$.
12. Repeat Steps 5 to 11 for $i \geq 1$.

Note that a queueing delay larger than δ_i in (2), or when $w_i = 0$ in (6), does not fail the measurement process detailed above. For example, when w_i = 0, the compound probe will arrive at dst with $M[G_i] = \frac{s_t}{c_n}$ to infer that the queueing delay at node i is either larger than or equal to the expected dispersion used for measurement.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCS.2020.2978231, IEEE Open Journal of the Computer Society

IV. Simulation Results

We implemented the proposed scheme using Python programming language [29] to study its performance in a simulated environment. Our rationale behind using computer simulation for evaluating the scheme is twofold. First, it is challenging to build or monitor a sustained queueing delay on a testbed or production network. Second, a large body of existing work on queueing delay have used simulation to evaluate their contributions [4], [14], [21], [22], [30].

In our study, we evaluated the proposed scheme using synthetically induced queueing delays on each router over end-to-end paths. This evaluation does not present a comparative performance against existing schemes because their feasibility and accuracy are tied to a strong network co-operation, as discussed in Section II; simulating their performance, therefore, seems futile. We discuss our simulation results on single- and multiple-hop paths below.

A. Single-Hop Path

We evaluated the proposed scheme over a single-hop path consisting of two 100 Mb/s links and one intermediate router (node 1), as Figure 6 shows. In a prior work, we performed an evaluation of the scheme over a similar path using ns-2 simulation [4]. Because this work implemented the scheme in Python code, single-hop evaluation here serves the following two purposes: i) it corroborates the utility of the proposed scheme for measuring queueing delay in an idealistic setup, as depicted in Figure 2, and ii) it confirms the correctness of the new source code that implements the proposed scheme.

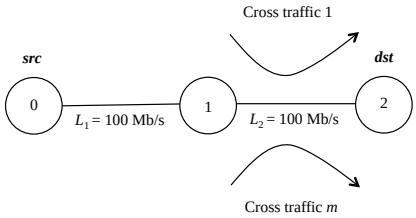


Fig. 6: Simulated single-hop path consisting of one router (node 1), which carries m, where $m > 1$, cross-traffic flows that build queueing delays over the path.

Figure 6 shows that we induced queueing delay on node 1 using multiple cross-traffic flows between node 1 and node 2. In our simulation, we modeled each of these flows as constant-bit rate (CBR) traffic [1] using 64-, 100-, and 200-byte packets to induce three different levels of queueing delay: light (2 μs $\pm$ 3 μs), moderate (4 μs $\pm$ 4 μs), and heavy (11 μs $\pm$ 9 μs), respectively. In these queueing delays, the first number is the average value whereas the second number is the standard deviation. Note that small queueing delays are available on the Internet [12] and the names used for the above-mentioned levels of queueing delay are specific only to our simulation setups regardless of any particular study on the Internet traffic.

For measuring queueing delay, we used $s_h = 64$ bytes (the smallest size for an IP packet) and $s_t = \{1000, 1500\}$ bytes in the compound probes. We did not use P_r in the probing structure since it is not a requirement in the single-hop scenario, as discussed in Section III. Table I presents the expected dispersion gaps in the compound probe (the last column) at node 1 for the used packet sizes according to (2). These values also correspond to the maximum queueing delays that are feasible for measurement using the selected packet sizes. For each s_t, we used 1000 compound probes[2], where each of them was separated from its adjacent probes with a random interval of 50 to 100 ms. This probing configuration allowed us to avoid self-interference among the probing packets and to capture the dynamic aspect of the induced queueing delays during measurement.

TABLE I: Expected dispersion gaps in the compound probe at node 1 over the single-hop path.

Probing-packet size		Link capacity		Dispersion gap
s_h (bytes)	s_t (bytes)	l_1 (Mb/s)	l_2 (Mb/s)	δ_1 (μs)
64	1000	100	100	74.88
64	1500	100	100	114.88

Figure 7 presents a summary of measurements on the single-hop path with (a) light, (b) moderate, and (c) heavy queueing delays, respectively, on node 1. Here, the x-axis of each graph refers to the size of the trailing packet and the y-axis refers to the queueing delay. The solid rectangle and the solid circle refer to the average of the induced (i.e., actual) and measured queueing delays, respectively, whereas the dashed and solid whiskers, respectively, refer to their standard deviations. In this paper, all queueing-delay values, both actual and measured, are rounded off to the next possible integer value in reference to the readily available clock resolution (i.e., 1μs) in OS kernels [31].

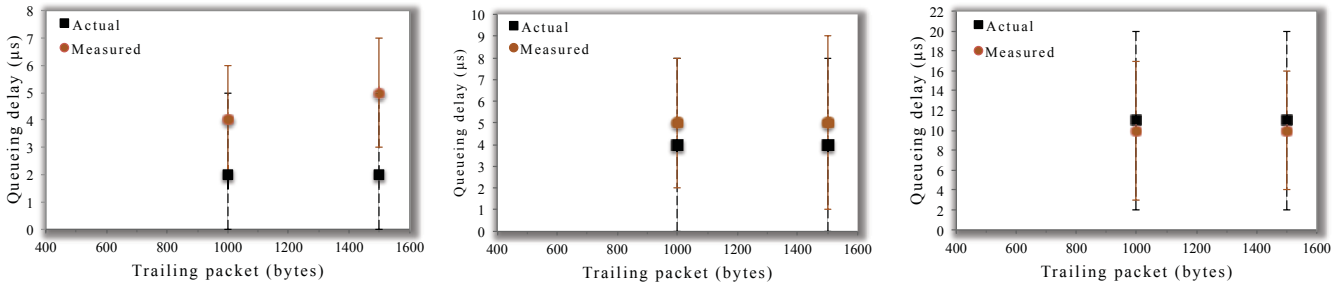
Figure 7(a) shows that the average queueing delays measured by the proposed scheme is not necessarily the same as the actual value for both packet sizes under the light queueing delay. However, there are strong overlaps between the ranges of the actual and measured values as identified by the corresponding whiskers in the graph. These overlaps are also prominent for both moderate and heavy queueing delays, as Figures 7(b) and 7(c) show, respectively. We interpret these strong overlaps as a reference to a high measurement accuracy for the proposed scheme since the average values (or any point estimates) provide a very little information about queueing delay that is dynamic in nature.

To provide a quantitative perspective of the above-stated performance, we calculated errors in the measured values from the degrees of overlaps between the actual and measured queueing delays. Here, error is defined as the ratio between the non-overlapping portion of the measured range and the total

[1]We use multiple CBR-traffic flows on the router to synthesize random queueing delays using the collision probabilities of the traffic packets for replicating the dynamic nature of the parameter of interest, as in [4]. The CBR flows in our simulation themselves do not represent the traffic pattern usually observed on the Internet.

[2]We chose 1000 compound probes from our prior experience with queueing-delay measurement in the single-hop scenario. [4]

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCS.2020.2978231, IEEE Open Journal of the Computer Society



(a) **node 1:** 2 μs $\pm$ 3 μs (b) **node 1:** 4 μs $\pm$ 4 μs (c) **node 1:** 11 μs $\pm$ 9 μs

Fig. 7: Measured values over the single-hop path with (a) light, i.e., 2 μs $\pm$ 3 μs, (b) moderate, i.e., 4 μs $\pm$ 4 μs, and (c) heavy, i.e., 11 μs $\pm$ 9 μs, queueing delays, respectively, on node 1.

range of the measured value. Under the light queueing delay, the proposed scheme has 25 and 50% errors when s_ts are 1000 and 1500 bytes, respectively. These errors are 0 and 12.5%, respectively, under the moderate queueing delay and 0% for both packet sizes under the heavy queueing delay. Overall, the average measurement error in Figure 7 is around 14%. This quantitative performance of the proposed scheme is in alignment with our prior evaluation using ns-2 simulation and the high accuracy justifies our motivation behind the single-hop simulation, as mentioned above in this section.

We also investigated the quality of data samples (i.e., measured intra-probe gaps at dst) used for determining the queueing delays over the single-hop path. Table II presents the percentage of valid data samples for the measured queueing delays in Figure 7. With valid data samples, we refer to those measured intra-probe gaps at dst that contained a compression. Note that the proposed scheme ignores intra-probe gaps with a zero compression or decompression (i.e., a larger than the expected dispersion) in its calculation, as described in (6).

TABLE II: Valid data samples (%) over the single-hop path.

Router (node i)	**node 1**	
s_t (bytes)	1000	1500
Light queueing delay	11.8	26.2
Moderate queueing delay	19.5	20.3
Heavy queueing delay	18.3	21.8

The second column of Table II shows around 12, 20 and 18% data samples were valid while measuring light, moderate, and heavy queueing delays, respectively, using s_t = 1000 bytes. These values for s_t = 1500 bytes (the last column) are around 26, 20, and 22%, respectively. This table once again corroborates the complexity of measuring queueing delay with the small proportions of valid data samples over the single-hop path.

B. Multiple-Hop Path

For multiple-hop measurement, we simulated the 4-hop path in Figure 8, as an extension to our most recent work on queuing delay [32]. This end-to-end path is made up of four 100 Mb/s links and three intermediate routers: node 1, node 2, and node 3, between src and dst. This figure shows multiple cross-traffic flows (CBR traffic with 64-, 100-, and 200-byte packets) were used to build queueing delays on each router over the path. For measuring queueing delay, we used s_h = 64 bytes and s_t = {800, 1000, 1200, 1500} bytes in the compound probes with appropriate sizes in P_r. As mentioned in Section IV-A, we once again used 1000 compound probes for each s_t with the same randomized separations (i.e., 50 to 100 ms) in order to avoid self-interference between adjacent probes.

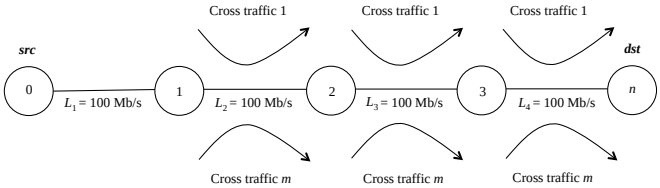


Fig. 8: Simulated 4-hop path consisting of three routers (node 1, node 2, and node 3), each of which carry m, where $m > 1$, cross-traffic flows that build queueing delays over the path.

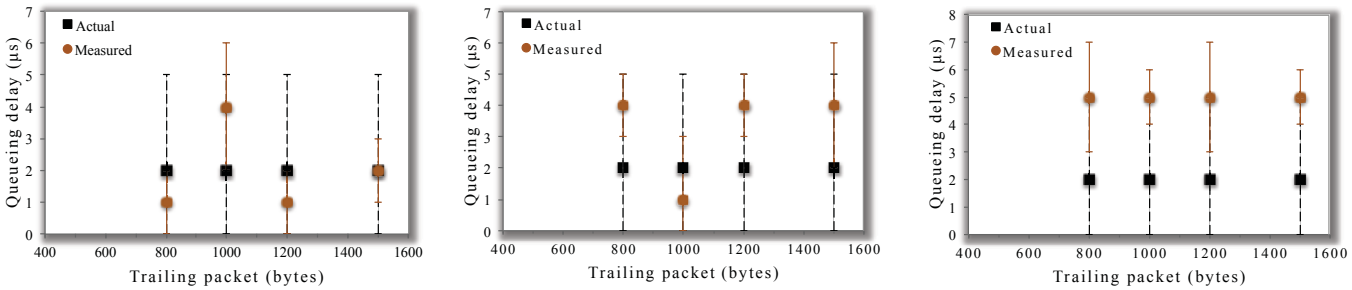
Table III shows the upper bounds of measurable queueing delays on node 1, node 2, and node 3 based on the expected dispersion gaps (the last column) for the selected s_ts. In reference to these bounds, we performed measurements on the multiple-hop path under two different traffic types: i) symmetric load and ii) asymmetric load. In the symmetric load, we induced identical queueing delays on the routers over the path whereas in the asymmetric load, we induced non-identical queueing delays on the routers over the path.

TABLE III: Expected dispersion gaps in the compound probe at nodes 1, 2, 3 over the multiple-hop path.

Probing-packet size		Link capacity		Dispersion gap
s_h (bytes)	s_t (bytes)	$l_{\{1,2,3\}}$ (Mb/s)	$l_{\{2,3,4\}}$ (Mb/s)	$\delta_{\{1,2,3\}}$ (μs)
64	800	100	100	58.88
64	1000	100	100	74.88
64	1200	100	100	90.88
64	1500	100	100	114.88

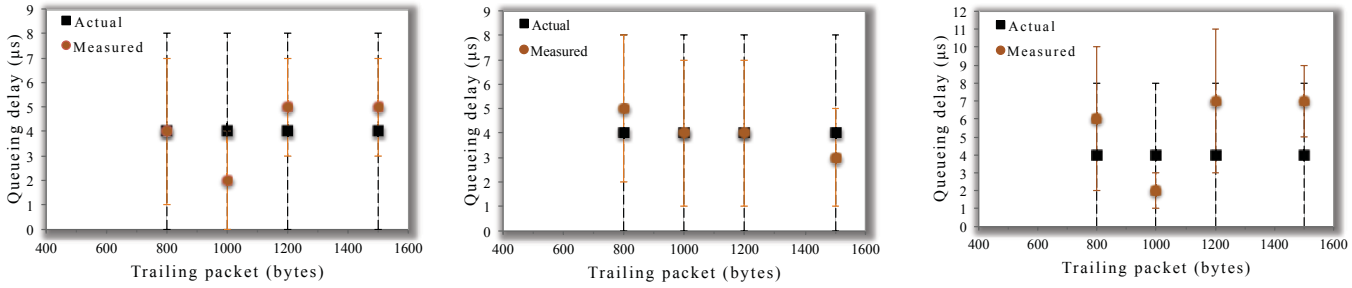
Under the symmetric load, we evaluated the proposed scheme with light, moderate, and heavy queueing delays, which we also used in the single-hop measurement. Figure 9

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCS.2020.2978231, IEEE Open Journal of the Computer Society



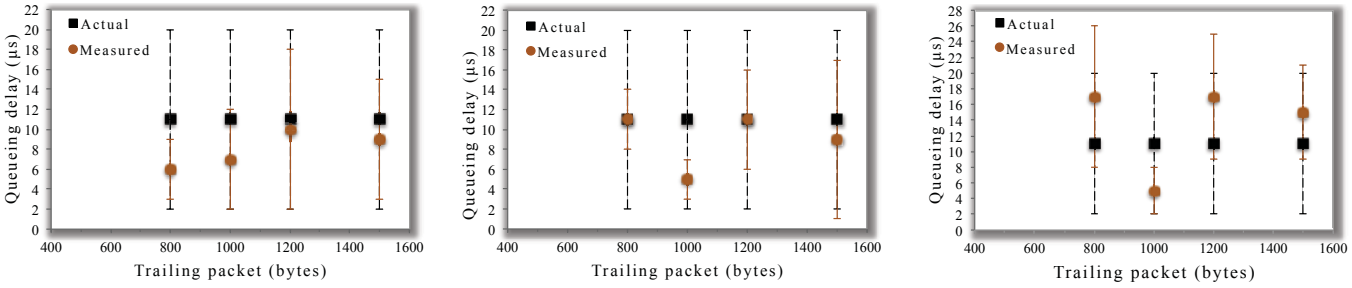
(a) **node 1:** 2 μs $\pm$ 3 μs (b) **node 2:** 2 μs $\pm$ 3 μs (c) **node 3:** 2 μs $\pm$ 3 μs

Fig. 9: Measured values over the multiple-hop path with symmetric & light queueing delay, i.e., 2 μs $\pm$ 3 μs, on (a) node 1, (b) node 2, and (c) node 3, respectively.



(a) **node 1:** 4 μs $\pm$ 4 μs (b) **node 2:** 4 μs $\pm$ 4 μs (c) **node 3:** 4 μs $\pm$ 4 μs

Fig. 10: Measured values over the multiple-hop path with symmetric & moderate queueing delay, i.e., 4 μs $\pm$ 4 μs, on (a) node 1, (b) node 2, and (c) node 3, respectively.



(a) **node 1:** 11 μs $\pm$ 9 μs (b) **node 2:** 11 μs $\pm$ 9 μs (c) **node 3:** 11 μs $\pm$ 9 μs

Fig. 11: Measured values over the multiple-hop path with symmetric & heavy queueing delay, i.e., 11 μs $\pm$ 9 μs, on (a) node 1, (b) node 2, and (c) node 3, respectively.

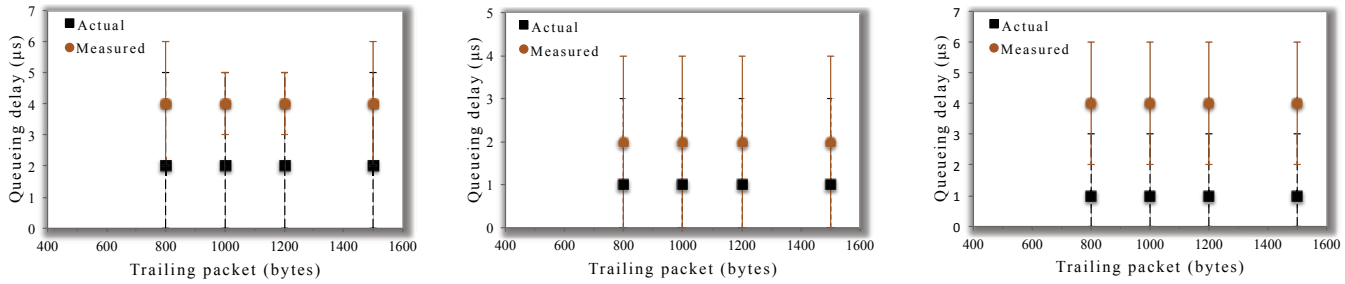
TABLE IV: Measurement error (%) under symmetric traffic over the multiple-hop path.

Router (node i)	**node 1**				**node 2**				**node 3**			
s_t (bytes)	800	1000	1200	1500	800	1000	1200	1500	800	1000	1200	1500
Light queueing delay	0	25	0	0	0	0	0	25	50	50	50	50
Moderate queueing delay	0	0	0	0	0	0	0	0	25	0	37.5	25
Heavy queueing delay	0	0	0	0	0	0	0	0	33.3	0	31.2	8.3

presents a summary of measurement under the symmetric load with light queueing delays (i.e., 2 μs $\pm$ 3 μs) on (a) node 1, (b) node 2, and (c) node 3, respectively. This figure shows that the ranges of the actual and measured values consistently overlap for each s_t, i.e., they present a high measurement accuracy, as in Section IV-A. Similarly, Figures 10 and 11 show that the measured values, overall, produce a high accuracy on each router over the path with moderate (i.e., 4 μs $\pm$ 4 μs) and heavy (i.e., 11 μs $\pm$ 11 μs) queueing delays, respectively.

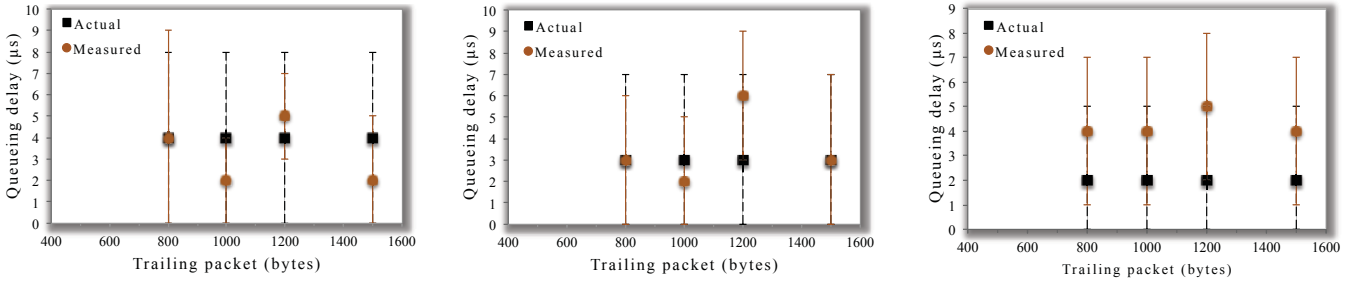
The quantitative error calculations for the measured values under the symmetric load considering the degrees of overlaps with the actual values in Figures 9–11 are presented in Table IV. This table shows 0% error in the majority of the 4-hop measurements irrespective of the used s_ts and the induced queueing delays. However, the scheme produce a relatively high error on node 3, especially with the light queueing delay.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCS.2020.2978231, IEEE Open Journal of the Computer Society



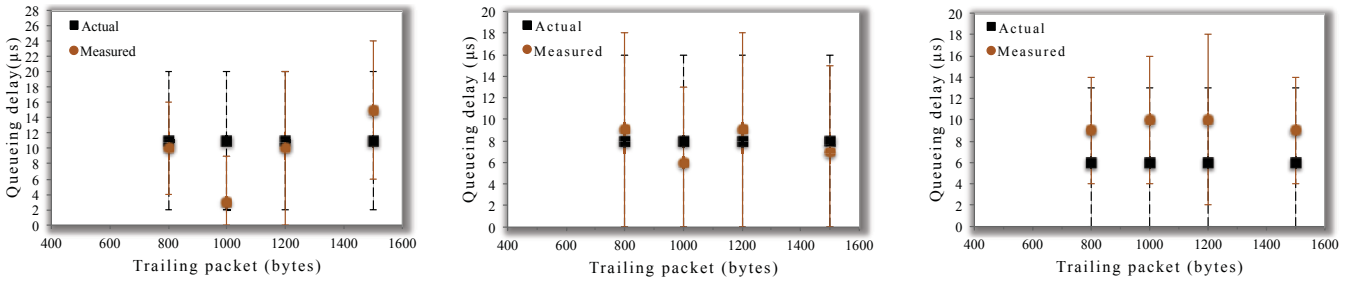
(a) **node 1:** 2 μs $\pm$ 3 μs (b) **node 2:** 1 μs $\pm$ 2 μs (c) **node 3:** 1 μs $\pm$ 2 μs

Fig. 12: Measured values over the multiple-hop path with asymmetric & light queueing delay, i.e., 2 μs $\pm$ 3 μs, 1 μs $\pm$ 2 μs, 1 μs $\pm$ 2 μs on (a) node 1, (b) node 2, and (c) node 3, respectively.



(a) **node 1:** 4 μs $\pm$ 4 μs (b) **node 2:** 3 μs $\pm$ 4 μs (c) **node 3:** 2 μs $\pm$ 3 μs

Fig. 13: Measured values over the multiple-hop path with asymmetric & moderate queueing delay, i.e., 4 μs $\pm$ 4 μs, 3 μs $\pm$ 4 μs, 2 μs $\pm$ 3 μs on (a) node 1, (b) node 2, and (c) node 3, respectively.



(a) **node 1:** 11 μs $\pm$ 9 μs (b) **node 2:** 8 μs $\pm$ 8 μs (c) **node 3:** 6 μs $\pm$ 7 μs

Fig. 14: Measured values over the multiple-hop path with asymmetric & heavy queueing delay, i.e., 11 μs $\pm$ 9 μs, 8 μs $\pm$ 8 μs, 6 μs $\pm$ 7 μs on (a) node 1, (b) node 2, and (c) node 3, respectively.

TABLE V: Measurement error (%) under asymmetric traffic over the multiple-hop path.

Router (node i)	**node 1**				**node 2**				**node 3**			
s_t (bytes)	800	1000	1200	1500	800	1000	1200	1500	800	1000	1200	1500
Light queueing delay	25	0	0	0	25	25	25	25	75	75	75	75
Moderate queueing delay	11.1	0	0	0	0	0	33.3	0	33.3	33.3	50	33.3
Heavy queueing delay	0	0	0	22.2	11.1	0	11.1	0	10	25	31.2	10

Such an error suggests two aspects previously observed in the single-hop results: i) measurement of a small queueing delay is challenging and ii) no specific s_t used in the simulation is conducive to a higher accuracy. In summary, the proposed scheme has an average measurement error of around 14% under the symmetric load.

Figures 12, 13, and 14 summarize the measured values under the asymmetric load with light, moderate, and heavy queueing delays, respectively. Here, we induced queueing delays of 2 μs $\pm$ 3 μs, 1 μs $\pm$ 2 μs, and 1 μs $\pm$ 2 μs on node 1, node 2, and node 3, respectively, to define the light queueing delay over the 4-hop path. For the moderate queueing delay, these values are 4 μs $\pm$ 4 μs, 3 μs $\pm$ 4 μs, and 2 μs $\pm$ 3 μs, respectively, and for the heavy queueing delay, they are 11 μs $\pm$ 9 μs, 8 μs $\pm$ 8 μs, and 6 μs $\pm$ 7 μs, respectively.

Under the asymmetric load, the measured values in the last

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCS.2020.2978231, IEEE Open Journal of the Computer Society

TABLE VI: Valid data samples (%) under symmetric traffic over the multiple-hop path.

Router (node i)	**node 1**				**node 2**				**node 3**			
s_t (bytes)	800	1000	1200	1500	800	1000	1200	1500	800	1000	1200	1500
Light queueing delay	6.4	8.6	14.0	5.8	4.4	35.5	13.3	4.9	29.4	19.7	28.8	18.8
Moderate queueing delay	12.7	16.3	14.5	7.1	7.3	36.7	10.7	8.8	32.8	15.4	27.0	17.2
Heavy queueing delay	6.5	16.6	9.4	8.1	2.3	5.2	10.5	10.0	21.5	11.5	19.0	10.5

TABLE VII: Valid data samples (%) under asymmetric traffic over the multiple-hop path.

Router (node i)	**node 1**				**node 2**				**node 3**			
s_t (bytes)	800	1000	1200	1500	800	1000	1200	1500	800	1000	1200	1500
Light queueing delay	11.1	10.4	18.0	6.9	31.1	21.4	30.9	22.2	21.6	17.7	19.2	14.7
Moderate queueing delay	28.4	16.5	13.6	26.6	24.4	19.6	9.9	22.5	21.7	26.2	18.0	21.1
Heavy queueing delay	16.1	11.1	19.6	16.3	20.8	17.3	28.3	25.8	23.7	30.0	25.4	23.6

three figures show a high accuracy in regards to the degrees of overlaps in each graph. This observation is also evident in the corresponding error calculations presented in Table V, where the average measurement error of the proposed scheme is around 20%. But the error values in Tables IV and V suggest that the asymmetric load, in comparison to the symmetric load, adds to the complexity of queueing-delay measurement.

We summarize the quality of the data samples used in the measurements under symmetric and the asymmetric loads in Tables VI and VII, respectively. Values in these tables show that compound probes experienced compression more frequently under the asymmetric load. However, such higher frequencies are uncorrelated with the corresponding measurement accuracies under the symmetric and asymmetric loads, as Tables IV and V summarize. This phenomenon reiterates the universal consensus on queueing delay as a complex network parameter for measurement on the Internet. Regardless of this challenge, the consistent and high accuracy of the proposed scheme with relatively small valid data samples, e.g., on average 14.6 and 20% in the last two tables, provide an insight into its efficacy of measurement.

V. Future Work

In general, the evaluation of the proposed scheme was successful. Our results suggest that the scheme is both accurate and sensitive to the used path lengths and different levels (e.g., light, moderate, and heavy) and types (e.g., symmetric and asymmetric) of traffic conditions during measurement. We consider that this study presents a thorough evaluation of the scheme in a controlled environment to initially establish its efficacy as a measurement tool. Therefore, further evaluation of this scheme concerning asymmetric link capacities with high transmission speeds (e.g., 100 Mb/s and above) over an end-to-end path, both in simulation and experimental environments, is an avenue for future work. In addition, we may look into the effect of packet processing time for determining the waiting time of packets in routers considering that queueing delay is defined by the competing traffic flows inside a router plus its hardware latencies at the same time. [33].

Another aspect we are keen on exploring for future research is the possibility of incorporating an advanced technique for filtering the data samples over the path. The proposed scheme currently uses simple statistics and only considers compression in the intra-probe gaps ignoring the decompressed values even though the later may contain useful information about queueing delay. Here, the motivation is to design an intelligent data-filtering technique (e.g., a machine-learning based solution) that will produce an enhanced measurement accuracy.

VI. Conclusions

Queueing delay is an important network parameter for determining the qualitative states of the Internet. Measurement of this parameter is a complex problem and existing schemes require a close co-operation from the path under test to alleviate the complexity of its measurement. In this paper, we proposed a scheme, called COMPRESS, to actively measure queueing delay on Internet routers over an end-to-end path. The proposed scheme uses compression in its UDP-based probing structure to determine queueing delay on all routers in a hop-by-hop manner. It is simple and self-sufficient in comparison to the existing schemes because it does not require infrastructural support and administrative access to the Internet for a successful measurement.

We studied the performance of the proposed scheme in a simulated environment using different queueing delays in single- and multiple-hop scenarios. The simulation results showed that the scheme is consistent and sensitive enough to produce a high accuracy regardless of the traffic conditions used in both scenarios. For example, the measured queueing delays always overlap with the actual values on the simulated paths and the average error across all measurements is around 20% or below. In the future, we will use this study as a motivation for evaluating the scheme using a more challenging path configurations and for improving the reported accuracy by incorporating an advanced data-filtering technique.

References

[1] R. Ramaswamy, N. Weng, and T. Wolf, "Characterizing network processing delay", in Proc. *IEEE Global Communications Conference*, TX, USA, 2004, pp. 1629–1634.

[2] K. Salehin, R. Rojas-Cessa, C. Lin, Z. Dong, and T. Kijkanjanarat, "Scheme to measure packet processing time of a remote host through estimation of end-link capacity," *IEEE Transactions on Computers*, vol. 64, no. 1, pp. 205–218, 2015.

[3] S. Laki, P. Mátray, P. Hága, I. Csabai, G. Vattay, "A model based approach for improving router geolocation," *Computer Networks*, vol. 54, no. 9, pp. 1490–1501, 2010.

[4] K. Salehin and R. Rojas-Cessa, "Scheme for measuring queueing delay of a router using probe-gap model: Single-hop case," *IEEE Communications Letters*, vol. 18, no. 4, pp. 696–699, 2014.

[5] A. Csoma, L. Toka, and A. Gulyas, "On lower estimating Internet queueing delay", in Proc. *IEEE International Conference on Telecommunications and Signal Processing*, Prague, Czech Republic, 2015, pp. 299–303.

[6] X. Chen, S. Feibish, Y. Koral, J. Rexford, O. Rottenstreich, S. Monetti, and T. Wang, "Fine-grained queue measurement in the data plane," in Proc. *ACM CoNEXT*, FL, USA, 2019, pp. 1–15.

[7] T. Høiland-Jørgensen, B. Ahlgren, P. Hurtig, and A. Brunstrom. "Measuring Latency Variation in the Internet," in Proc. *ACM International Conference on Emerging Networking Experiments and Technologies*, NY, USA, 2016, pp. 473–480.

[8] B. Gueye, A. Ziviani, M. Crovella, and S. Fdida, "Constraint-based geolocation of Internet hosts," *IEEE/ACM Transactions on Networking*, vol. 14, no. 6, pp. 1219–1232, 2006.

[9] E. Katz-Bassett, J. John, A. Krishnamurthy, D. Wetherall, T. Anderson, and Y. Chawathe, "Towards IP geolocation using delay and topology measurements," in Proc. *ACM Internet Measurement Conference*, NY, USA, 2006, pp. 71–84.

[10] K. Salehin, V. Sahasrabudhe, and R. Rojas-Cessa, "Determination of interrupt-coalescence latency of remote hosts through active measurement," *IEEE Access*, vol. 6, pp. 1–15, 2018.

[11] G. Fettweis, "The tactile Internet: applications and challenges," *IEEE Vehicular Technology Magazine*, vol. 9, no. 1, pp. 64–70, 2014.

[12] K. Papagiannaki, S. Moon, C. Fraleigh, P. Thiran, and C. Diot, "Measurement and analysis of single-hop delay on an IP backbone network," *IEEE Journal on Selected Areas on Communications*, vol. 21, no. 6, pp. 908–921, aug. 2003.

[13] L. Angrisani, G. Ventre, L. Peluso, and A. Tedesco, "Measurement of processing queueing delays introduced by an open-source router in a single-hop network," *IEEE Transactions on Instrumentation and Measurement*, vol. 55, no. 4, pp. 1065–1076, 2006.

[14] M. Garetto and D. Towsley, "Modeling, simulation and measurements of queueing delay under long-tail Internet traffic," in Proc. *ACM SIGMETRICS*, NY, USA, 2003, pp. 47–57.

[15] B. Choi, S. Moon, Z. Zhang, K. Papagiannaki, and C. Diot, "Analysis of point-to-point packet delay in an operational network", *Computer Networks*, vol. 51, no. 3, pp. 3812–3827, 2007.

[16] V. Cerf, V. Jacobson, N. Weaver, and J. Gettys, "Bufferbloat: What's wrong with the Internet?," *Communications of the ACM*, vol. 55, no. 2, pp. 40–47, 2012.

[17] C. Chirichella and D. Rossi, "To the Moon and back: Are Internet bufferbloat delays really that large?," in Proc. *IEEE International Conference on Computer Communications*, Turin, Italy, 2013, pp. 3297–3302.

[18] M. Haiyan, Y. Jinyao, P. Georgopoulos, and B. Plattner, "Towards SDN based queueing delay estimation," *China Communications*, vol. 13, no. 3, pp. 27–36, 2016.

[19] V. Jacobson. Pathchar – A tool to infer characteristics of Internet paths. [Online]. Available: ftp://ftp.ee.lbl.gov/pathchar/msri-talk.pdf.

[20] A. Downey, "Using pathchar to estimate internet link characteristics," in Proc. *ACM SIGCOMM*, MA, USA, 1999, pp. 241–250.

[21] N. Duffield, J. Horowitz, F. Presti, and D. Towsley, "Network delay tomography from end-to-end unicast measurements," in Proc. *Thyrrhenian International Workshop on Digital Communications*, Taormina, Italy, 2001, pp. 576–595.

[22] A. Adams, T. Bu, T. Friedman, J. Horowitz, D. Towsley, R. Caceres, N. Duffield, F. Presti, S. Moon, and V. Paxson, "The use of end-to-end multicast measurements for characterizing internal network behavior," *IEEE Communications Magazine*, vol. 38, no. 5, pp. 152–159, 2000.

[23] K. Anagnostakis, M. Greenwald, and R. Ryger, "cing: measuring network-internal delays using only existing infrastructure," in Proc. *IEEE International Conference on Computer Communications*, 2003, pp. 2112–2121.

[24] K. Salehin and R. Rojas-Cessa, "A combined methodology for measurement of available bandwidth and link capacity in wired packet networks," *IET Communications*, vol. 4, no. 2, pp. 240–252, 2010.

[25] K. Salehin and R. Rojas-Cessa, "Packet-pair sizing for controlling packet dispersion on wired heterogeneous networks," in Proc. *IEEE International Conference on Computing, Networking and Communications*, CA, USA, 2013, pp. 1031–1035.

[26] K. Lai and M. Baker, "Measuring link bandwidths using a deterministic model of packet delay," in Proc. *ACM SIGCOMM*, Stockholm, Sweden, 2000, pp. 283–294.

[27] K. Harfoush, A. Bestavros, and J. Byers, "Measuring bottleneck bandwidth of targeted path segments," in Proc. *IEEE International Conference on Computer Communications*, CA, USA, 2003, pp. 2079–2089.

[28] B. Mah. pchar: A tool for measuring Internet path characteristics. [Online]. Available: https://www.kitchenlab.org/www/bmah/Software/pchar/.

[29] Welcome to Python.org. Accessed August 15, 2018. [Online]. Available: https://www.python.org/.

[30] N. Duffield and F. Presti, "Network tomography from measured end-to-end delay covariance," *IEEE/ACM Transactions on Networking*, vol. 12, no. 6, pp. 978–992, 2004.

[31] L. De Vito, S. Rapuano and L. Tomaciello, "One-way delay measurement: State of the art," in *IEEE Transactions on Instrumentation and Measurement*, vol. 57, no. 12, pp. 2742–2750, 2008.

[32] K. Salehin, R. Rojas-Cessa, and K. Kwon, "COMPRESS: A self-sufficient scheme for measuring queueing delay on the Internet routers," in Proc. *IEEE International Conference on Computing, Networking, and Communications*, HI, USA, 2019, pp. 624–629.

[33] B. Briscoe et al., "Reducing Internet latency: A survey of techniques and their merits," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 2149–2196, 2016.