

Robust Available Bandwidth Estimation Against Dynamic Behavior of Packet Scheduler in Operational LTE Networks

Takashi Oshiba, Kousuke Nogami, Koichi Nihei, and Kozo Satoda
System Platform Research Laboratories, NEC Corporation, Japan

Abstract—We shed new light on the mechanism behind how the dynamic behavior of a packet scheduler at the link layer in mobile networks degrades the accuracy of conventional available bandwidth (i.e., unused capacity of an end-to-end path) estimation methods that use a probing packet train (i.e., a set of multiple probing packets). Most of the conventional methods, which were originally designed for wired networks, estimate available bandwidth at the receiver by detecting changes of the observed queuing delays of probing packets. They utilize a microscopic approach in which they check the difference of the queuing delay of each packet on a packet-by-packet basis in order to detect the queuing delay changes. We found that the dynamic behavior of a packet scheduler at the link layer dramatically disturbs the queuing delays observed at the receiver. The disturbed queuing delays make it tremendously difficult for the conventional microscopic approach to detect changes of the delays, resulting in degraded estimation accuracy.

As a countermeasure, we propose a method called PathQuick3 for robustly estimating the available bandwidth against the dynamic behavior of a packet scheduler at the link layer in mobile networks. We developed a robust algorithm that utilizes a macroscopic approach, extracting change trends from the widely fluctuating queuing delays throughout a packet train.

Our experimental evaluation over an operational LTE network showed that PathQuick3 clearly outperformed a conventional method in terms of accuracy and stability of estimated values. In the best case, the estimation error and the variability of the values estimated by PathQuick3 were only 8.5% and 18.9% that of the conventional method, respectively.

Keywords—available bandwidth; LTE link layer; packet scheduler; curve fitting; nonlinear least squares method

I. INTRODUCTION

Smartphones with LTE connectivity have superseded PCs with wired connectivity as the dominant way of Internet access [1] due to the recent diffusion of LTE [2] and the rise of smartphones [3]. Nowadays, ordinary smartphone users utilize bandwidth-intensive applications such as on-demand video streaming (e.g., YouTube, Netflix, Hulu), live video streaming (e.g., Ustream, Periscope), video chat (e.g., Skype, Apple FaceTime, Google Hangouts), and photo sharing (e.g., Instagram, Snapchat, Pinterest) over LTE networks. In order to ensure Quality of Experience (QoE) for these bandwidth-intensive applications, the accurate measurement of available bandwidth (i.e., physical capacity minus bandwidth being used during a certain time period [4]) is of great importance. This is because, for example, in a video streaming system, the system can adapt video bit-rate dynamically [5] on the basis of the

currently available bandwidth. This can realize video streaming without playback interruption, thus ensuring QoE.

However, the accurate measurement of available bandwidth over LTE networks remains a challenge. In this paper, we explore the reason behind the difficulty and show that dynamic behavior of the packet scheduler at the link layer in LTE networks degrades the accuracy of conventional available bandwidth estimation methods that use a probing packet train (i.e., a set of multiple probing packets). More specifically, as we explain in detail in Section IV-B, packet scheduling for every 1 ms at a base station [6] dramatically disturbs *queuing delays observed at a receiver* (the definition is given in Section III-B-3). This disturbance severely affects the queuing delays of all probing packets, making it one of the main causes of the accuracy degradation.

In this paper, as a countermeasure, we propose a method called PathQuick3 for robustly estimating the available bandwidth against the disturbance of queuing delays. To this end, PathQuick3 utilizes the idea of curve fitting along with the nonlinear least squares method [7]. PathQuick3 is the successor to our previous methods, PathQuick [8] and PathQuick2 [9].

We focus on the downlink direction of LTE networks in this paper because the traffic volume of the downlink can be more than ten times as large as that of the uplink [10], and thus the traffic volume of the downlink is dominant in LTE networks.

The main contributions of this paper are fourfold:

- (1) We found that the dynamic behavior of the packet scheduler at the LTE link layer is one of the main causes of accuracy degradation of available bandwidth estimation.
- (2) To the best of our knowledge, we elucidated, for the first time, the mechanism of how the packet scheduler severely disturbs queuing delays observed at a receiver.
- (3) We developed an available bandwidth estimation method that is *robust* against the highly disturbed queuing delays.
- (4) To the best of our knowledge, we conducted, for the first time, experimental evaluations of available bandwidth estimation over an *operational LTE network in the wild*.

II. RELATED WORK

A. Prior Work for Wired Networks

Much prior work has been done on end-to-end available bandwidth estimation that actively sends probing packet trains [4]. Representative examples include pathChirp [11], Pathload [12], and PTR [13]. These methods and our PathQuick and PathQuick2 were originally designed for wired networks.

According to an experimental comparison in Wi-Fi networks [14], pathChirp has better estimation accuracy than Pathload and PTR. While this result pertains to Wi-Fi networks, pathChirp has the potential for accurate estimation over LTE networks too, so we compare pathChirp with PathQuick3 in Section VI.

B. Prior Work for Wi-Fi Networks

After the prior work for wired networks received much attention in the research community, many methods for Wi-Fi networks were proposed [15]–[19]. Most of them depend heavily on the Wi-Fi protocol (e.g., CSMA/CA, exponential backoff waiting time, half-duplex communication, and MAC frame aggregation) in order to leverage their estimation accuracy. Although they have been evaluated in Wi-Fi networks, they have not yet been studied in mobile networks, except WBest [15] (see below).

C. Prior Work for Mobile Networks

Available estimation over mobile networks has not been studied extensively. Although Wi-Fi networks and mobile networks are both wireless, accurate available estimation over mobile networks using prior works originally designed for Wi-Fi networks is challenging because mobile networks behave differently from Wi-Fi networks. Actually, while WBest was recently evaluated over the downlink direction of a (nowadays, relatively old-fashioned) 3G 1xEV-DO network in [20], the authors of [20] concluded that WBest is practically infeasible for the 3G network due to inaccurate results. Note that the detailed mechanism of how the inaccurate results occurred was not analyzed in [20]. Although M-PFProbe [21] was also recently evaluated over the downlink direction of a 3G 1xEV-DO network, the estimation accuracy was not revealed—only the estimated values were shown. While [22] mentioned mobile networks, evaluations were conducted over wired networks only, not mobile networks.

To the best of our knowledge, thus far there has been no research in which evaluation over the 4G LTE network, with either a simulator or an operational network, has been conducted.

III. OVERVIEW OF AVAILABLE BANDWIDTH ESTIMATION

A. Basic Principle: Probe Rate Model

We begin with a description of one of the basic principles of available bandwidth estimation, called the Probe Rate Model (PRM) [23]. PRM has been broadly utilized by prior work, including pathChirp, Pathload, PTR, and our own PathQuick and PathQuick2. In PRM, a sender transmits a UDP packet train to a receiver. The receiver then estimates the available bandwidth and reports the estimated result to the sender. PRM is based on the observation that (a) if the probing rate of a packet train at a sender is less than the available bandwidth, the probing packets will face no queuing delay inside the network, so the time interval for each probing packet observed at a receiver will be the same as at the sender. On the other hand, (b) if the probing rate exceeds the available bandwidth, the packets will be queued inside the network, increasing the time intervals observed at the receiver. The available bandwidth can be estimated by observing the probing rate at which there is a transition from (a) to (b).

B. Probe Rate Model in PathQuick

1) Design of Packet Train Structure of PathQuick

As a concrete example of the PRM principle, we explain the estimation mechanism of PathQuick. We designed the packet train structure of PathQuick for short estimation duration and probing over a wide range of rates as follows. Note that PathQuick2 and PathQuick3 utilize the same packet train structure. In order to keep the whole transmission duration of a packet train short, the time interval for each packet within the packet train must be short. To this end, we designed the packet train so that each packet is placed at an equal time interval (see Fig. 1-(1)). Also, in order to probe over a wide range of rates with a single packet train, the per-packet probing rate must be changed within the single packet train. To this end, we designed the structure so that each packet size linearly increases from the previous one as the packet sequence proceeds (see Fig. 1-(2)).

Let us consider a packet train consisting of N probing packets. Each packet within the packet train is placed at equal time interval T_{quick} at the sender (Fig. 1-(1)). The whole transmission duration of a packet train (i.e., the packet train length) is

$$T_{train}^{(quick)} = T_{quick} \cdot (N-1) = T_{quick} \cdot N - T_{quick}. \quad (1)$$

For simplicity, we omit the transmission (or serialization) delay of each probing packet from $T_{train}^{(quick)}$. In practical terms, this omission does not matter because the transmission delay is usually sufficiently shorter than T_{quick} ; i.e., short enough to be ignored. Thus, packet train length $T_{train}^{(quick)}$ is a linear function of the number of probing packets N . This $O(N)$ nature enables PathQuick to keep the packet train length short.

The packet size of each probing packet is

$$P_i = P_1 + (i-1) \cdot \Delta P = \Delta P \cdot i + (P_1 - \Delta P), \quad (2)$$

where $i=1,2,\dots,N$ and the constant value ΔP is the increase amount of the packet size (Fig. 1-(2)). Thus, each packet size P_i is a linear function of i , since P_1 and ΔP are constant values.

The per-packet probing rate at the i -th packet—i.e., the momentary probing rate of the packet train—is

$$R_i = \frac{P_i}{T_{quick}} = \frac{\Delta P}{T_{quick}} i + \frac{P_1 - \Delta P}{T_{quick}}. \quad (3)$$

Thus, each per-packet probing rate R_i is also a linear function of i . Therefore, PathQuick can increase the per-packet probing rate within a single packet train, and thereby can probe over a wide range of rates using a single packet train.

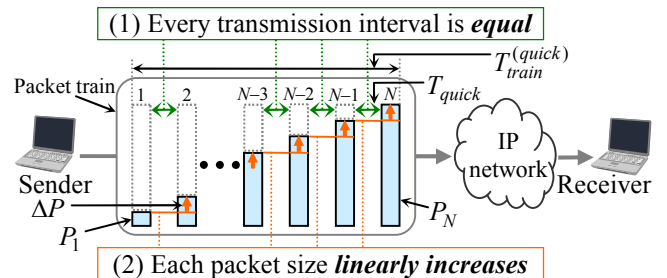


Fig. 1. Design of packet train structure.

2) PRM-based Available Bandwidth Estimation

Let us define the time interval between the $(i-1)$ -th and i -th packet observed at the receiver as T_i^{rcv} .

The receiver analyzes the observed time intervals based on the PRM principle to estimate the available bandwidth as follows:

$$(a) T_i^{rcv} = T_{quick}, \quad \text{if } R_i \leq A \quad (4)$$

$$(b) T_i^{rcv} < T_i^{rcv}, \quad \text{otherwise,}$$

where A is the actual available bandwidth, i.e., the ground truth.

In PathQuick, a per-packet probing rate $R_k = P_k / T_{quick}$, where the k -th packet is the packet at which the observed time intervals at receiver T_{k+1}^{rcv} begin increasing, becomes the estimated available bandwidth. Figure 2 illustrates the meaning of Eq. (4), $T_2^{rcv} = T_3^{rcv} = \dots = T_{k-1}^{rcv} = T_k^{rcv} = T_{quick}$ in (a) and $T_{quick} < T_{k+1}^{rcv} < T_{k+2}^{rcv} < \dots < T_N^{rcv}$ in (b). That is, the k -th packet is the transition point of PRM, and the per-packet probing rate of the k -th packet becomes the estimated available bandwidth.

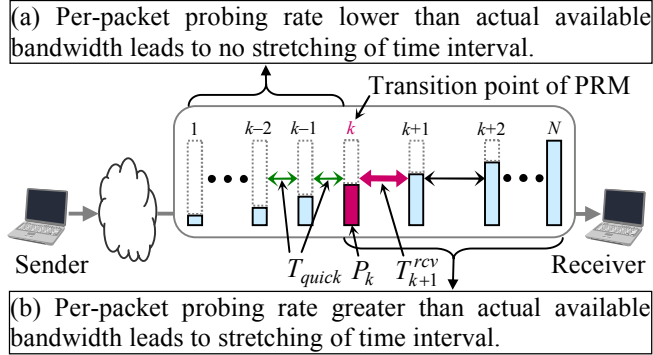


Fig. 2. PRM-based available bandwidth estimation.

3) Queuing Delays Observed at a Receiver

As discussed above, finding the right transition point is essential in PRM. Here, we explain in detail how to find it in PathQuick. In PathQuick, the transition point is identified by using the *queuing delay of each packet observed at a receiver* q_i ($i=1,2,\dots,N$). q_i is defined as

$$q_i = (r_i - r_1) - (s_i - s_1), \quad (5)$$

where r_i is the receiving time of i -th packet in the receiver clock and s_i is the transmission time of the i -th packet in the sender clock. Namely, q_i is the duration between the receiving time of the 1st packet and that of the i -th packet minus the duration between the transmission time of the 1st packet and that of the i -th packet (see Fig. 3).

In the case of Fig. 3, the transition point k is 3, $q_2 = q_3 = 0$ and $0 < q_4 < q_5 < q_6$. Thus, Eq. (4) can be rewritten as

$$(a) q_i = 0, \quad \text{if } R_i \leq A \quad (6)$$

$$(b) q_i > 0, \quad \text{otherwise.}$$

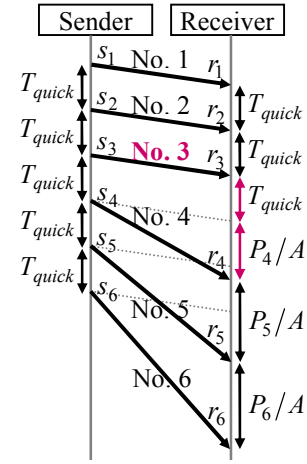


Fig. 3. Transmission time and receiving time of each probing packet. The explanation of P_i/A is given in Section V-A.

IV. IMPACT OF DYNAMIC BEHAVIOR OF PACKET SCHEDULER AT LTE LINK LAYER

In wired networks, two of the main causes of degraded estimation accuracy with conventional methods are time-varying cross-traffic and multiple bottlenecks [24]. In LTE networks, the main causes of the degradation are not only the time-varying cross-traffic and multiple bottlenecks but also the dynamic behavior of the packet scheduler at the LTE link layer.

A. Overview of Packet Scheduler in LTE Downlink

Here, we outline the dynamic behavior of the packet scheduler in LTE downlink. In LTE downlink, multi-path fading, interference with neighboring cells, and path loss (i.e., attenuation of signal) due to propagation distance lead to rapid time-varying radio quality of the wireless channel [25]. Moreover, due to the mobility of user equipment (UE), the number of UEs in the cell of a base station (i.e., an Evolved Node B, or eNB) is also time-varying. Many operational LTE networks utilize a packet scheduler (e.g., a proportional fair scheduler [26][27]) that takes the time-varying radio channel quality and number of UEs into account [6]. The packet scheduler periodically assigns radio resources, called resource blocks, and transmits packets with the assigned resource block to UEs for every Transmission Time Interval (TTI). In LTE networks, the TTI is 1 ms [6].

B. Disturbance of Queuing Delays by a Packet Scheduler

The 1-ms TTI means that the packet scheduler repeats (1) buffering and (2) transmission every 1 ms. Namely, it (1) buffers incoming packets from a wired packet core network and (2) transmits the buffered multiple packets at once, i.e., in a *bursty* manner. This behavior enables eNBs to dynamically adapt to the time-varying nature of the LTE network, and is effective for high throughput, low latency, and fairness among UEs. However, we found this behavior to be quite harmful from the point of view of available bandwidth estimation using a probing packet train. In short, this repetitious *stop-and-go* or *ON-OFF* behavior injects strong *burstiness* into the probing packets, resulting in severe disturbance of queuing delays observed at a receiver (i.e., q_i).

Figure 4-(ii), a conceptual example, illustrates what happens when a packet train arrives at an eNB. We assume all per-packet probing rates are less than the actual available bandwidth, i.e., $R_i < A$. We also assume the time interval for each packet is 0.25 ms, and thus $s_i = s_1 + 0.25 \times (i-1)$. This means, if the size of a packet is 1,500 bytes, the probing rate of the packet is $8 \times 1,500 / 0.25 = 48$ Mbps. Since the 48-Mbps bandwidth is comparable to today's LTE downlink, the 0.25-ms time interval is a realistic assumption. At the eNB, the single packet train is split into multiple chunks, and multiple probing packets (four in this case) are collected on each chunk. At the receiver, the four packets arrive at the same time, so $r_2 = r_3 = r_4 = r_5$ and $r_6 = r_7 = r_8 = r_9$. Let us calculate the queuing delays of several packets with Eq. (5):

$$\begin{aligned} q_2 &= (r_2 - r_1) - (s_2 - s_1) = (r_2 - r_1) - ((s_1 + 0.25 \times 1) - s_1) = 1 - 0.25 \times 1 = 0.75, \\ q_5 &= (r_5 - r_1) - (s_5 - s_1) = (r_5 - r_1) - ((s_1 + 0.25 \times 4) - s_1) = 1 - 0.25 \times 4 = 0, \\ q_6 &= (r_6 - r_1) - (s_6 - s_1) = (r_6 - r_1) - ((s_1 + 0.25 \times 5) - s_1) = 2 - 0.25 \times 5 = 0.75, \end{aligned}$$

and thus $(q_2, q_3, q_4, q_5) = (q_6, q_7, q_8, q_9) = (0.75, 0.50, 0.25, 0)$.

Note that this means cyclic expansion and contraction of queuing delays, i.e., cyclic ((b), (b), (b), (a)) in Eq. (6). In contrast, in a wired network (Fig. 4-(i)), $q_i = 0$ for all probing packets. This means it is always (a) in Eq. (6).

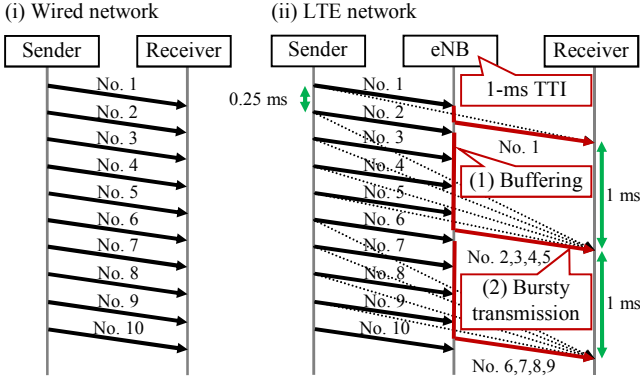


Fig. 4. Disturbance of queuing delays in LTE networks. The black and red solid arrows show the actual packet flow, i.e., white box view to network nodes, while the gray dotted arrows show the packet flow in an end-to-end black box view to network nodes.

Figure 5-(i) shows queuing delays of ten packet trains of PathQuick obtained from a private wired network without cross-traffic. We designed the packet trains so that the per-packet probing rates near the tail of the packet train exceeded the actual available bandwidth. There is no cross-traffic, so each of the ten queuing delays are quite similar. Since these ten queuing delays strictly agree with Eq. (6), one can determine the exact transition point of PRM easily: it is the 46th or 47th packet, depending on the packet trains.

Figure 5-(ii) shows the queuing delays of a single packet train of PathQuick obtained from an operational LTE network. We designed the packet trains so that the per-packet probing rates near the head of the packet train exceeded the actual available bandwidth. In contrast to Fig. 5-(i), due to the repetitious stop-and-go behavior of the packet scheduler, the shape of the queuing delays is like the teeth of a saw, and thus the position of the transition point of PRM seems quite indistinct. If we use Eq. (6), the transition point of PRM becomes the first packet; however, the estimated value

$R_1 = P_1 / T_{quick}$ ends up considerably underestimated compared to the actual available bandwidth. Although the above examples are of PathQuick, this phenomenon can be generally applied to other conventional PRM-based methods such as pathChirp. Consequently, in the presence of the dynamic behavior of a packet scheduler in LTE networks, the accuracy of conventional PRM-based methods severely degrades.

To make matters worse, the height of the seven vertical jumps in Fig. 5-(ii) is around 5 ms, and thus it is often not 1 ms (like Fig. 4-(ii)) but more than 1 ms. It seems that since we obtained the data in Fig. 5-(ii) at a crowded area, with many other UEs in the same cell, our UE was not assigned a resource block for every 1-ms TTI but rather for every 5 ms. The height may change depending on crowdedness (and other conditions), so when designing a new estimation algorithm, one must not assume the fixed 1-ms height.

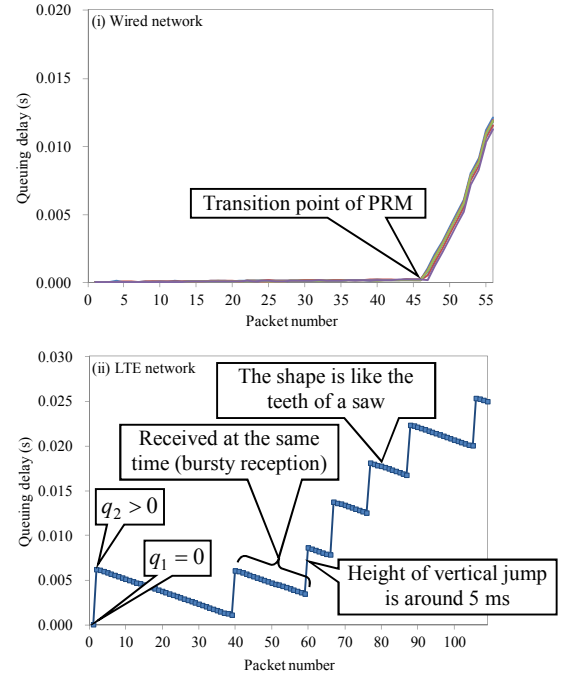


Fig. 5. Observed queuing delays at a receiver in (i) a wired network and (ii) an LTE network.

V. PROPOSAL OF PATHQUICK3

Here, we propose a method, PathQuick3, for robustly estimating the available bandwidth against the disturbance of queuing delays in the presence of the dynamic behavior of a packet scheduler in LTE networks.

A. Ideal State Queuing Delay Model

The observations shown in Fig. 5 led us to perceive that even though the detailed shape of queuing delays in Fig. 5-(i) and (ii) are different at the microscopic view, both of them are similar at the macroscopic view. Namely, in Fig. 5-(i), the queuing delays move horizontally halfway and then move on to the upper direction. Similarly, in Fig. 5-(ii), the chunks of queuing delays move horizontally halfway and then move on to the upper direction. Thus, noise-free (Fig. 5-(i)) and noisy (Fig. 5-(ii)) queuing delays demonstrate a similar trend. We

therefore utilize the idea of *curve fitting* in order to fit a noise-free model to noisy queuing delays.

We constructed a model of noise-free (i.e., ideal state) queuing delays called the Ideal state Queuing delays Model (IQM). Let us explain our IQM with Fig. 3. Recall the transition point $k = 3$. If $i < 3$, $R_i < A$ and thus $q_i = 0$. When $i = k = 3$, $A = R_3 = P_3/T_{quick}$.

When $i = k + 1 = 4$, $R_4 > A$. This means that a probing packet with a packet size of P_4 arrives at a bottleneck network pipe whose unused capacity is A , and the per-packet probing rate R_4 exceeds the unused capacity A . The fourth packet is then queued, and the queuing duration is P_4/A . Thus, $r_4 - r_1 = 3T_{quick} + P_4/A$. Next, when $i = 5$, the fifth packet arrives at the pipe while the fourth packet is still being queued. After the fourth packet is transmitted, the fifth packet must additionally wait for P_5/A . Thus, using Eq. (5) and Fig. 3,

$$\begin{aligned} q_4 &= (r_4 - r_1) - (s_4 - s_1) = (3T_{quick} + P_4/A) - (3T_{quick}) = P_4/A, \\ q_5 &= (r_5 - r_1) - (s_5 - s_1) = (3T_{quick} + P_4/A + P_5/A) - (4T_{quick}) \\ &= P_4/A + P_5/A - T_{quick}, \\ q_6 &= (r_6 - r_1) - (s_6 - s_1) = (3T_{quick} + P_4/A + P_5/A + P_6/A) - (5T_{quick}) \\ &= P_4/A + P_5/A + P_6/A - 2T_{quick}. \end{aligned}$$

If $i > k$, we can generalize as

$$\begin{aligned} q_i &= \frac{1}{A}(P_{k+1} + P_{k+2} + \dots + P_i) - \{i - (k+1)\}T_{quick} \\ &= \frac{1}{A}\{(P_1 + \dots + P_{k+1} + \dots + P_i) - (P_1 + \dots + P_k)\} - \{i - (k+1)\}T_{quick} \\ &= \frac{1}{A}\left(\sum_{j=1}^i P_j - \sum_{j=1}^k P_j\right) - \{i - (k+1)\}T_{quick} \\ &= \frac{1}{A}\left[\frac{i}{2}2P_1 + (i-1)\Delta P\right] - \{i - (k+1)\}T_{quick} \\ &= \frac{1}{A}\left[P_1(i-k) + \frac{\Delta P}{2}\{i(i-1) - k(k-1)\}\right] - \{i - (k+1)\}T_{quick}. \end{aligned}$$

With $A = R_k = P_k/T_{quick}$,

$$q_i = \frac{T_{quick}}{P_k}\left[P_1(i-k) + \frac{\Delta P}{2}\{i(i-1) - k(k-1)\}\right] - \{i - (k+1)\}T_{quick}.$$

Thus, Eq. (6) can be rewritten as

$$\begin{aligned} \text{(a) } q_i &= 0, & \text{if } i \leq k \\ \text{(b) } q_i &= \frac{T_{quick}}{P_k}\left[P_1(i-k) + \frac{\Delta P}{2}\{i(i-1) - k(k-1)\}\right] \\ &\quad - \{i - (k+1)\}T_{quick}, & \text{otherwise.} \end{aligned} \quad (7)$$

Hence, q_i in (b) is a quadratic function of i . Thus, Eq. (7) means that queuing delay q_i is a nonlinear function of i , where (a) a horizontal line (if $i \leq k$) and (b) a parabola (if $i > k$) are connected at joint point k . We call the nonlinear curve an *ideal curve* since the queuing delays in Fig. 3 are noise-free (i.e., there is no burstiness caused by cross-traffic or a packet scheduler). To visualize and explain our IQM, we show the ideal curves where joint point k moves on from left (i.e., $k = 1$) to right (i.e., $k = N$) in Fig. 6. The number of ideal curves is N . The shape of the ideal curves is ruled by the estimation parameters of PathQuick (i.e., P_1 , ΔP , N , and T_{quick}). The observed queuing delays in Fig. 5 are also shown in Fig. 6.

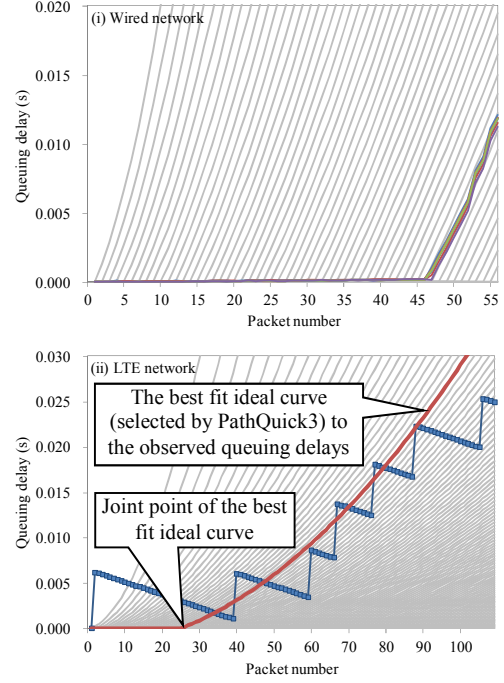


Fig. 6. Ideal curves of queuing delays (gray) of IQM, overlapped with observed queuing delays in (i) wired network and (ii) LTE network.

B. Curve Fitting and Nonlinear Least Squares Method

In Fig. 6-(i), an ideal curve whose joint point is the 46th or 47th packet is the best fit to the observed queuing delays. This ensures that Eq. (7) can accurately draw the shape of the observed (noise-free) queuing delays.

Our eyes can easily recognize that the joint point of the best fit ideal curve is the 46th or 47th packet from more than 50 candidates of joint points. But how can a machine (i.e., not a human) automatically do this? To find out, we utilize the idea of *curve fitting* along with the *nonlinear least squares method* [7]. Let us define observed queuing delays as Q_i ($i = 1, 2, \dots, N$). Actually, q_i in Eq. (7) is a multivariate function, i.e., its variables are k and i , since other parameters such as P_1 , ΔP , and T_{quick} are explicitly given by a PathQuick3 user. Therefore, we replace q_i with $q(k, i)$ and rewrite Eq. (7) as

$$\begin{aligned} \text{(a) } q(k, i) &= 0, & \text{if } i \leq k \\ \text{(b) } q(k, i) &= \frac{T_{quick}}{P_k}\left[P_1(i-k) + \frac{\Delta P}{2}\{i(i-1) - k(k-1)\}\right] \\ &\quad - \{i - (k+1)\}T_{quick}, & \text{otherwise.} \end{aligned} \quad (8)$$

In order to determine the best fit ideal curve, we find out an ideal curve that has the minimum sum of squared errors between the observed queuing delays Q_i and the ideal curve $q(k, i)$. Since $q(k, i)$ is a nonlinear function, this type of problem can be solved by a nonlinear least squares method in curve fitting in computational statistics [7]. However, most solvers for these are complex and compute-intensive, so we choose instead to utilize a simple and intuitive method as follows. Let us define the sum of squared errors between the observed queuing delays Q_i and the ideal curve $q(k, i)$ for each k as $SSE(k)$ and define the joint point of the best fit ideal curve as k_{best} . Then, $SSE(k)$ and k_{best} can be computed in a light-weight manner, even on smartphones, as follows:

$$SSE(k) = \sum_{i=1}^N \{Q_i - q(k, i)\}^2, \quad (9)$$

$$k_{best} = \underset{k}{\operatorname{arg\,min}} SSE(k).$$

Now, we can systemically find out the joint point of the best fit ideal curve with k_{best} . For example, the k_{best} is the 25th packet, as the corresponding best fit ideal curve is drawn with a bold red curve in Fig. 6-(ii). As we can see, the selected ideal curve visually fits the observed queuing delays quite well, even if the degree of disturbance of the observed queuing delays is more severe than that of Fig. 6-(i). Finally, we combine the IQM concept with the PRM principle and estimate the available bandwidth with PathQuick3 \hat{A} as

$$\hat{A} = R_{k_{best}} = P_{k_{best}} / T_{quick}. \quad (10)$$

Our nonlinear least squares approach takes a macroscopic view, i.e., it can extract a changed trend from the widely fluctuated queuing delays throughout the packet train and systemically find out the best fit ideal curve in terms of the sum of squared errors. Consequently, PathQuick3 can robustly estimate the available bandwidth against the disturbance of queuing delays over LTE networks.

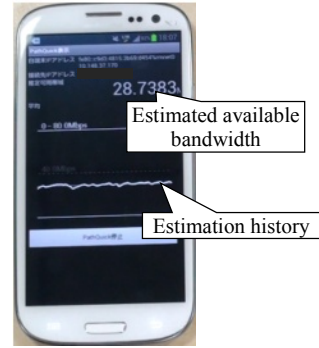


Fig. 7. Snapshot of the prototype PathQuick3 receiver.

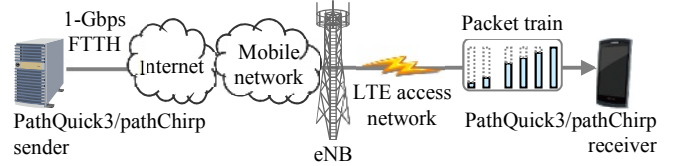


Fig. 8. Experimental environment over an operational LTE network.

VI. EXPERIMENTAL EVALUATION OVER AN OPERATIONAL LTE DOWNLINK NETWORK

We conducted an experimental evaluation of PathQuick3, in terms of accuracy and variability of estimated available bandwidth, over an operational LTE downlink network of Japan's primary mobile operator.

A. Experimental Setup

1) Ground Truth of Available Bandwidth

Since we cannot access the network nodes of the mobile operator directly, the *ground truth* of the available bandwidth is unknown to us. Instead, although available bandwidth and bulk TCP throughput are not the same network metric [24], we follow [20] as our precedent and treat bulk TCP throughput as a *reference* to the ground truth (or *best effort* ground truth [20]). We obtained bulk TCP throughput with one of the most famous speed test applications in Japan [28].

2) Comparison with pathChirp and System Setup

We investigated the public availability of the source code of the prior work for Wi-Fi networks [15]–[19] and found that only the source code of WBest is publicly available. However, as we mentioned in Section II-C, WBest has been reported as infeasible in mobile networks. We therefore chose pathChirp for comparison (as mentioned in Section II-A).

We implemented a prototype PathQuick3 receiver as an Android application (see Fig. 7). The pathChirp source code for UNIX [29] was ported into the Android OS. We used an Android smartphone (octa-core 2.1 GHz+1.5 GHz CPU, 3 GB RAM, Android OS 5.0.2) [30] for the experimentation. A Linux server (quad-core 1.7 GHz CPU, 8 GB RAM, Ubuntu 14.04) with a 1-Gbps FTTH connection was deployed for the PathQuick3 and pathChirp senders (see Fig. 8).

The experiment was performed at eight diverse locations in Tokyo, the most populous metropolitan area in the world [31]. At each location, we (1) ran a downlink speed test once, (2) received 30 probing packet trains of pathChirp, and (3) received 30 probing packet trains of PathQuick3.

3) Estimation Parameter Choice

We chose the probable bandwidth range of both methods as follows. The current fastest average LTE downlink speed by country in the world is 36 Mbps [32], and the LTE downlink speed has continued to increase in recent years. With an eye to the future, we doubled the rate, choosing 72 Mbps as the maximum probable bandwidth of a probing packet train of both methods.

To realize the 72-Mbps target of maximum probable bandwidth with PathQuick3, we set the packet size of the first packet $P_1 = 36$ bytes, the increase of the amount of packet size $\Delta P = 13$ bytes, the number of packets in a packet train $N = 109$, and the equal time interval $T_{quick} = 0.16$ ms. Therefore, the packet size of the last packet $P_N = 36 + 13 \times (109 - 1) = 1,440$ bytes and so the maximum probable bandwidth is $P_N / T_{quick} = 8 \times 1,440 / (0.16 \times 10^{-3}) = 72$ Mbps. Then, the minimum probable bandwidth with PathQuick3 becomes $P_2 / T_{quick} = 8 \times (36 + 13) / (0.16 \times 10^{-3}) = 2.5$ Mbps.

In pathChirp, the maximum and minimum probable bandwidth is set to the same as PathQuick3 with $-u$ and $-l$ options [29], respectively. Since the recommended packet size of pathChirp is more than 1,000 bytes [11], we set the packet size of all packets in the packet train of pathChirp to the same as that of the last packet of PathQuick3, i.e., 1,440 bytes. Although pathChirp uses a moving average filter [29] to smooth multiple estimated available bandwidths, since we would like to collect raw values (i.e., not smoothed) of the estimated available bandwidth from each packet train, we disabled the moving average filter.

B. Experimental Results

The means of estimated available bandwidth from 30 probing packet trains of PathQuick3 and pathChirp are shown in Fig. 9. The standard deviation error bars (i.e., $\pm 1\sigma$) of the 30 probing packet trains are also shown. Overall, our PathQuick3 clearly outperformed pathChirp in terms of both accuracy and variability. Specifically, all of the mean estimated values of PathQuick3 are closer to the reference values than those of

pathChirp, demonstrating that PathQuick3 is more accurate, and all of the standard deviation error bars of PathQuick3 are shorter than those of pathChirp, which indicates that PathQuick3 can output more stable estimates than pathChirp.

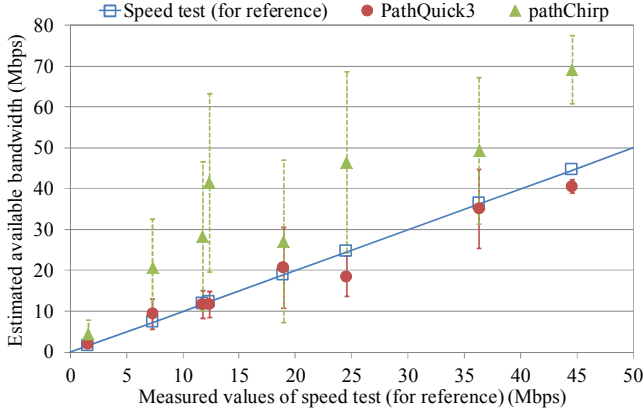


Fig. 9. The means of estimated available bandwidth \pm standard deviation at eight locations in Tokyo. Sample size: $n = 30$.

1) Estimation Accuracy

In order to analyze the estimation accuracy of both methods in detail, the Mean Absolute Error (MAE) of both methods is shown in Fig. 10. The MAE is defined as

$$MAE = \frac{1}{n} \sum_{i=1}^n |\hat{A} - A| \quad (11)$$

where \hat{A} is estimated available bandwidth, A is the measured value of the speed test, and n is the sample size of \hat{A} ($n = 30$ in this case).

The MAE of PathQuick3 was consistently less than that of pathChirp. The ratio of MAE (i.e., the MAE of PathQuick3 divided by that of pathChirp, expressed as a percentage) is also shown. The ratio of MAE ranged from 8.5% to 49.5%. Thus, the estimation error of PathQuick3 is more than one order of magnitude smaller than pathChirp in the best case, and the estimation error of PathQuick3 is less than a half of pathChirp even in the worst case.

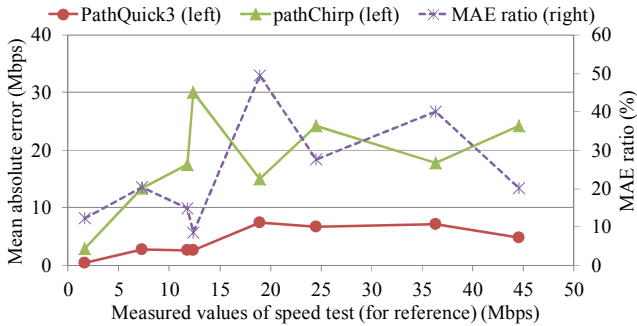


Fig. 10. The mean absolute errors of estimated available bandwidth.

2) Variability of Estimated Values

In order to analyze the variability of estimated values of both methods in detail, the Coefficient of Variation (CV) of both methods is shown in Fig. 11. CV is a measure of

dispersion of data relative to the mean. It is defined as $CV = \sigma/\mu$, where σ is the standard deviation of the 30 estimated values and μ is the mean of that. We use CV because it is useful for comparing the variability of samples of data when the means are different (as we can see in Fig. 9 that the means have various values).

The CV of PathQuick3 is consistently less than that of pathChirp. The ratio of CV (i.e., the CV of PathQuick3 divided by that of pathChirp, expressed as a percentage) is also shown. In the best case, the ratio of CV is only 18.9%. Thus, the variability of estimated values of PathQuick3 is smaller than pathChirp.

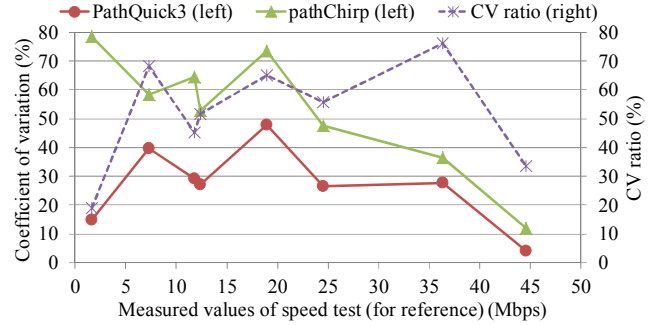


Fig. 11. The coefficient of variations of estimated available bandwidth.

3) Observed Queuing Delays in the Wild

Examples of observed queuing delays of the 30 packet trains of (i) PathQuick3 and (ii) pathChirp are shown in Fig. 12.

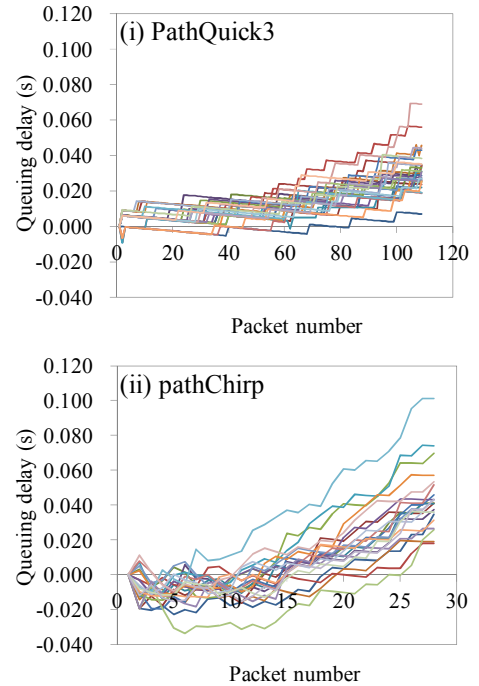


Fig. 12. Observed queuing delays of 30 packet trains at one location. Each colored curve corresponds to queuing delays of a single packet train.

In PathQuick3, the shapes of queuing delays are like the teeth of a saw, as we saw in Fig. 5-(ii). All of the queuing delays have different shapes. This implies that each packet train experienced diverse effects of bursty cross-traffic,

repetitious stop-and-go behavior of the packet scheduler, and queuing behavior at network nodes. Judging from the accurate results of Fig. 10, our IQM successfully extracted changed trends from such noisy queuing delays and identified the correct transition point of PRM.

In pathChirp, noisy queuing delays were also observed. Judging from the inaccurate results of Fig. 10, we conclude that its estimation algorithm, which was meant to check the difference of queuing delay of each packet on a packet-by-packet basis (see Fig. 3 of [11]), failed to identify the correct transition point of PRM.

4) Intrusiveness of Measurement

We compared the total packet size of a single packet train (i.e., intrusiveness) using both methods. The intrusiveness of PathQuick3 was $36 + 49 + \dots + 1,440 = 80.4$ kB. We observed 28 packets in a single packet train of pathChirp, and thus the intrusiveness of pathChirp was $28 \times 1,440 = 40.3$ kB. Hence, the difference of intrusiveness between both methods is 40.1 kB. We believe this difference is acceptable for typical smartphone users since they consume several gigabytes of data volume per month [10] that is five orders of magnitude larger than the difference of intrusiveness. We plan to evaluate the PathQuick3 with less intrusiveness as our future work.

VII. CONCLUSION AND FUTURE WORK

PathQuick3 is a method that can robustly estimate the available bandwidth against the disturbance of queuing delays. Our experimental evaluation over an operational LTE downlink network has shown that PathQuick3 clearly outperformed pathChirp in terms of accuracy and stability of estimated values. In the best case, the estimation error and the variability of estimated values of PathQuick3 were only 8.5% and 18.9% that of pathChirp, respectively.

We are currently conducting experiments over an operational LTE *uplink* network, and preliminary results are encouraging. In future work, we plan to conduct *large-scale* experiments over operational LTE (and emerging LTE-Advanced) networks of multiple mobile operators.

REFERENCES

- [1] comScore, Inc., "Mobile Internet usage skyrockets in past 4 years to overtake desktop as most used digital platform," April 13, 2015. [online] <https://www.comscore.com/Insights/Blog/Mobile-Internet-Usage-Skyrockets-in-Past-4-Years-to-Overtake-Desktop-as-Most-Used-Digital-Platform>
- [2] GSA, "480 LTE networks commercially launched in 157 countries," January 26, 2016. [online] http://gsacom.com/wp-content/uploads/2016/02/160216-LTE_world_map_480_networks_launched.jpg
- [3] F. Qian et al., "How to reduce smartphone traffic volume by 30%?," *PAM Conference*, pp. 42–52, 2013.
- [4] R. Prasad, C. Dovrolis, M. Murray, and K. Claffy, "Bandwidth estimation: Metrics, measurement techniques, and tools," *IEEE Network*, Vol. 17, Issue 6, pp. 27–35, 2003.
- [5] S. Akhshabi, A. C. Begen, and C. Dovrolis, "An experimental evaluation of rate-adaptation algorithms in adaptive streaming over HTTP," *ACM MMSys*, pp. 157–168, 2011.
- [6] F. Capozzi, G. Piro, L. A. Grieco, G. Boggia, and P. Camarda, "Downlink packet scheduling in LTE cellular networks: Key design issues and a survey," *IEEE Communications Surveys & Tutorials*, Vol. 15, Issue 2, pp. 678–700, 2013.

- [7] C. T. Kelley, "Iterative methods for optimization," *SIAM Frontiers in Applied Mathematics*, No. 18, 1999.
- [8] T. Oshiba and K. Nakajima, "Quick end-to-end available bandwidth estimation for QoS of real-time multimedia communication," *IEEE ISCC*, pp. 162–167, 2010.
- [9] T. Oshiba and K. Nakajima, "Quick and simultaneous estimation of available bandwidth and effective UDP throughput for real-time communication," *IEEE ISCC*, pp. 1123–1130, 2011.
- [10] Nokia Solutions and Networks White Paper, "What is going on in mobile broadband networks?," 2014. [online] http://networks.nokia.com/system/files/document/nokia_smartphone_traffic_white_paper.pdf
- [11] V. J. Ribeiro, R. H. Riedi, R. G. Baraniuk, J. Navratil, and L. Cottrell, "pathChirp: Efficient available bandwidth estimation for network paths," *PAM Workshop*, 2003.
- [12] M. Jain and C. Dovrolis, "End-to-end available bandwidth: Measurement methodology, dynamics, and relation with TCP throughput," *ACM SIGCOMM*, pp. 295–308, 2002.
- [13] N. Hu and P. Steenkiste, "Evaluation and characterization of available bandwidth probing techniques," *IEEE JSAC*, Vol. 21, No. 6, pp. 879–894, 2003.
- [14] D. Gupta et al., "Experimental comparison of bandwidth estimation tools for wireless mesh networks," *IEEE INFOCOM*, pp. 2891–2895, 2009.
- [15] M. Li, M. Claypool, and R. Kinicki, "WBEST: A bandwidth estimation tool for IEEE 802.11 wireless networks," *IEEE LCN*, pp. 374–381, 2008.
- [16] M. Portoles-Comeras et al., "Impact of transient CSMA/CA access delays on active bandwidth measurements," *ACM IMC*, pp. 397–409, 2009.
- [17] A. Farshad et al., "On the impact of 802.11n frame aggregation on end-to-end available bandwidth estimation," *IEEE SECON*, pp. 108–116, 2014.
- [18] A. Johnsson and M. Björkman, "On measuring available bandwidth in wireless networks," *IEEE LCN*, pp. 861–868, 2008.
- [19] H. K. Lee et al., "Bandwidth estimation in wireless LANs for multimedia streaming services," *IEEE ICME*, pp. 1181–1184, 2006.
- [20] D. Koutsonikolas and Y. C. Hu, "On the feasibility of bandwidth estimation in wireless access networks," *Wireless Networks*, Vol. 17, Issue 6, pp. 1561–1580, 2011.
- [21] U. Devi et al., "On the estimation of available bandwidth in broadband cellular networks," *IEEE SECON*, pp. 19–27, 2014.
- [22] C. U. Castellanos et al., "Comparison of available bandwidth estimation techniques in packet-switched mobile networks," *IEEE PIMRC*, pp. 1–5, 2006.
- [23] L. Lao, C. Dovrolis, and M. Y. Sanadidi, "The probe gap model can underestimate the available bandwidth of multihop paths," *ACM SIGCOMM CCR*, Vol. 36, Issue 5, pp. 29–34, 2006.
- [24] M. Jain and C. Dovrolis, "Ten fallacies and pitfalls on end-to-end available bandwidth estimation," *ACM IMC*, pp. 272–277, 2004.
- [25] A. Larmo et al., "The LTE link layer design," *IEEE Communications Magazine*, Vol. 47, Issue 4, pp. 52–59, 2009.
- [26] K. Winstein, A. Sivaraman, and H. Balakrishnan, "Stochastic forecasts achieve high throughput and low delay over cellular networks," *USENIX NSDI*, pp. 459–471, 2013.
- [27] Q. Xu, S. Mehrotra, Z. M. Mao, and J. Li, "PROTEUS: Network performance forecast for real-time, interactive mobile applications," *ACM MobiSys*, pp. 347–360, 2013.
- [28] IID, Inc., "RBB TODAY SPEED TEST," *Google Play*. [online] <https://play.google.com/store/apps/details?id=com.rbbtoday.speedtest>
- [29] V. J. Ribeiro et al., pathChirp source code version 2.4.1. [online] <http://www.spin.rice.edu/Software/pathChirp/pathchirp-2.4.1.tar.gz>
- [30] Samsung Electronics Co., Ltd., "Samsung Galaxy S6," <http://www.samsung.com/global/galaxy/galaxys6/galaxy-s6/>
- [31] Wikipedia, "List of metropolitan areas by population." [online] https://en.wikipedia.org/wiki/List_of_metropolitan_areas_by_population
- [32] OpenSignal, "The state of LTE (September 2015), LTE speed by country." [online] <http://opensignal.com/reports/2015/09/state-of-lte-q3-2015/>