Available Bandwidth Estimation via One-Way Delay Jitter and Queuing Delay Propagation Model*

Yu-Chen Huang[1,2], *Chun-Shien Lu*[1,†] *Hsiao-Kuang Wu*[2]
[1]*Institute of Information Science, Academia Sinica, Taipei, Taiwan, ROC*
[2]*Dept. Computer Sci. & Information Eng., National Central Uni., Chung-Li, Taiwan, ROC*

Abstract

We propose a one-way delay jitter based scheme, "JitterPath," for available bandwidth estimation. Common assumptions, including use of the fluid traffic model and use of the bottleneck link capacity, that have been made in the literature are relaxed in this study. We exploit one-way delay jitter and accumulated queuing delay to predict the type of a queuing region for each packet pair. In addition, we quantify the captured traffic ratio, which is defined as the total output gaps of joint queuing regions per total input gaps, and use it to derive the relationship between probing rate and available bandwidth. We further investigate how the estimation resolution and the probing noise ratio are related to the accuracy of available bandwidth estimation. Extensive simulations and real-network experiment have been conducted and comparisons with other methods have been made to verify the effectiveness of our method, no matter whether single-hop or multi-hop environments are considered.

keywords: Available bandwidth, Bottleneck, One-Way-Delay jitter, Probing noise, Queuing delay/region

1 Introduction

Active probing has been found to be useful for discovering network conditions. In [8], Keshav proposed a new idea, called the "packet pair," to estimate the capacity of a bottleneck link, which is defined as the minimum capacity among the links from the sender to the receiver. However, the capacity of a bottleneck link is not really decisive to affect network conditions. On the contrary, available bandwidth [5] determined from all the links as the minimum bandwidth that has not been used has been recognized as the most important factor. Thus, it is known that a bottleneck link is not necessarily the link with the minimum available bandwidth.

In order to actively estimate available bandwidth, probing packets are sent and interact with cross traffic such that the relationship between available bandwidth and probing rate can be revealed. The existing active probing-based available bandwidth estimation technologies can be roughly divided into two categories [18]: (1) probe gap model (PGM) methods and (2) probe rate model (PRM) methods.

The idea behind PRM-based methods [3, 6, 9, 11, 16] is to exploit self-induced congestion, which relies on an intuitive assumption that if the probing rate is higher than the available bandwidth, then the probing packets will start to be queued in the bottleneck router, resulting in increasing one-way delay. PRM methods assume the cross traffic operates like a fluid traffic model [7] so that the arrival rates are constant and the amount of cross traffic inserted between each packet pair is the same. Under these circumstances, if the probing rate is higher than the available bandwidth, then the queuing delays of probing packets will constitute an increasing and smoothing sequence; otherwise, queuing delay will not occur. Unfortunately, TCP is bursty instead of fluid, so OWD increasing trend does not suffice to represent the relationship between the probing rate and available bandwidth, especially, when the cross traffic rate varies during measurement.

In PGM-based methods [1, 13, 15, 18], probing packets are sent into the network in the hope that the inter-departure time can be different from the inter-arrival time at the receiver side, and this information can be exploited to measure available bandwidth. Under this circumstance, if the amount of the cross traffic placed in between the packet pair can be known, then it is possible to directly estimate the available bandwidth as the bottleneck link capacity minus the cross traffic rate without needing to pour many probing trains to the network. PGM methods make two assumptions: (1) a single-hop environment is assumed such that the link with the minimum available bandwidth is the bottleneck link; (2) the ideal fluid cross traffic model is used.

*This research was supported by the National Science Council under NSC grants 93-2213-E-001-026 and 94-2213-E-001-027.
†Contact author (C. S. Lu, email: lcs@iis.sinica.edu.tw)

1-4244-0270-0/06/$20.00 (c)2006 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2006 proceedings.

Our contribution is that a new method, JitterPath, is proposed based on the exploitation of one-way delay jitter for available bandwidth. JitterPath contains a strategy called the "queuing delay propagation model" that can be used to determine the type of a queuing region that exists within a packet pair. JitterPath also defines the "captured traffic ratio," which is the ratio of the total output gaps accumulated from JQRs per total input gaps and is used to determine the relationship between the probing rate and available bandwidth. There are three major characteristics of our method: (1) JitterPath can work without assuming the use of a fluid traffic model; (2) Since JitterPath does not rely on the use of the bottleneck link's capacity to indicate the relationship between the probing rate and available bandwidth, it is feasible in a multi-hop environment; and (3) JitterPath alleviates the impact of probing noises under bursty cross traffic model.

2 Proposed Method, JitterPath, Under Fluid Traffic Model

In this section, we first describe the proposed method under the fluid cross traffic model, in which probing noise does not exist and the packet pairs of the same packet train operate in the same queuing region.

2.1 Queuing Delay Propagation Model

The queuing region (QR) determines the utilization of bottleneck router during the inter-arrival time of a packet pair. A QR can be classified into two types: joint queuing region (JQR) and disjoint queuing region (DQR). More specifically, a JQR is strictly defined as the situation where the bottleneck router is fully busy to service aggregated traffic during the time interval between a packet pair, which also means that the bottleneck router is never idle during that interval. The aggregated traffic is the sum of the probing and cross traffics.

2.1.1 Queuing Region Determination

Let PP^i $(1 \leq i \leq n)$, which is composed of consecutive probing packets P^{i-1} and P^i, denote the i-th packet pair in the j-th packet train, whose inter-departure time (the input gap between a packet pair) is Δ_{in}^{j}. Let QR^i denote the queuing region of PP^i. If PP^i is defined as operating in a JQR, then the bottleneck router does not finish three tasks before the arrival of P^i: processing the queued traffic (let Q^{i-1} denote the sum of queued traffic and undelivered packets when P^{i-1} arrives at the router), processing P^{i-1} (let s be the constant packet size), and processing the cross traffic inserted between this packet pair PP^i (let $CT_{\Delta_{in}}^{i}$ denote the amount of cross traffic arriving the bottleneck during the inter-arrival time of PP^i). They can be related to each other as

$$\left(\frac{Q^{i-1} + s + CT_{\Delta_{in}}^{i}}{TC}\right) > \Delta_{in}^{j}, \quad \text{if } QR^i \text{ is a JQR}, \quad (1)$$

where TC denotes the capacity of a tight-link. Based on the definition of a JQR, the inter-arrival time of a packet pair PP^i measured at the receiver side is $\frac{s+CT_{\Delta_{in}}^{i}}{TC}$, which is the output gap, Δ_{out}^{i}. In addition, let Q^{i-1}/TC be denoted as D^{i-1} representing the queuing delay that accumulates before packet P^{i-1}. Thus, Eq. (1) can be rewritten as

$$\Delta_{out}^{i} > (\Delta_{in}^{j} - D^{i-1}), \quad \text{if } QR^i \text{ is a JQR}. \quad (2)$$

Eq. (2) can be further rewritten by using the one-way delay jitter to substitute the difference between the output and input gaps as

$$\Delta OWD^i + D^{i-1} > 0, \quad \text{if } QR^i \text{ is a JQR}, \quad (3)$$

where $\Delta OWD^i = \Delta_{out}^{i} - \Delta_{in}^{j}$.

As shown in Eq. (3), we define what can be derived if a packet pair operates in a JQR. In practice, in order to determine the queuing region type of a packet pair, what the receiver needs based on Eq. (3) is the OWD jitter and the accumulated queuing delay. In addition, OWD jitter is also known to closely depend on accumulated queuing delay, which implies that an accurate queuing delay propagation mechanism is indispensable. We shall discuss this issue in Sec. 2.1.2.

Under the situation of the fluid traffic model, the condition that can be used to determine a queuing region to be a JQR is defined as

$$QR^i \text{ is JQR, if } \Delta OWD^i + D^{i-1} > 0. \quad (4)$$

In Eq. (4), $\Delta OWD^i + D^{i-1} > 0$ reveals that the packet pair PP^i captures a certain amount of cross traffic. However, the queuing region QR^i of PP^i is not certain to be a JQR because the captured cross traffic is not guaranteed to completely fill the gap between PP^i. If $\Delta OWD^i + D^{i-1} > 0$ and there is room exists within PP^i, then the output gap is a DQR but is erroneously determined to be a JQR. This phenomenon is called an "estimation error," which is caused by the probe noise and will be further described in more detail in Sec. 3 by taking bursty traffic model into consideration.

2.1.2 Queuing Delay Propagation

Once QR^i has been determined based on Eq. (4), the accumulated queuing delay D^i needs to be calculated so that the next queuing region can be determined. Depending on whether PP^i operates in a JQR or not, the queuing delay that accumulates in front of packet P^i can be

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2006 proceedings.

derived as

$$D^i = \begin{cases} D^{i-1} + \Delta OWD^i, \text{ if } QR^i \text{ is JQR} \\ max(0, \Delta OWD^i), \text{ if } QR^i \text{ is DQR.} \end{cases} \quad (5)$$

As a result, if the procedures shown in Eqs. (4) and (5) are iteratively performed, queuing region determination and queuing delay propagation can be accomplished. In addition, it is also obvious that the queuing delay propagation and the one-way delay jitter are two key factors. However, it should be noted that D^0, the "initial queuing delay," is the queuing delay in front of the first probing packet in a packet train and is still unknown. In the next subsection, we explain how D^0 can be determined.

2.1.3 Determining the Initial Queuing Delay (D^0)

As described in Sec. 2.1, the accumulated queuing delay and one-way delay jitter play key roles in determining the type of a queuing region. Both of them are exploited here to determine the initial queuing delay. Our method only needs to trace the output gaps of a packet train once for initial queuing delay determination. We will investigate this issue based on two cases.

First, we consider the case where the OWD jitter is less than zero, i.e., $\Delta OWD^i < 0$. This means that the first output gap is decreased to absorb the initial queuing delay. The size of the first output gap depends on two factors: the queuing delay and cross traffic. If there is no cross traffic, or if the amount of inserted cross traffic is not large enough to expand the first output gap, then $|\Delta OWD^1|$ is equal to the initial queuing delay D^0. On the other hand, if $D^0 > |\Delta OWD^1|$, then subsequent output probing gaps will be gradually decreased to absorb D^0. Propagation of the queuing delay will continue, and a decreasing sequence of queuing delays, $D^0 > D^1 > \cdots > D^{k-1}$, will be generated until $D^{k-1} < D^k$ that is caused by the inserted cross traffic is satisfied. This means that the initial queuing delay will be completely exhausted. Therefore, the initial queuing delay can be calculated and found to be the absolute value of D^{k-1}, $|D^{k-1}|$.

In the second case, we consider the OWD jitter is larger than zero, i.e., $\Delta OWD^i > 0$. In this situation, we cannot be sure if the initial queuing delay will be completely absorbed in the first output gap or if it does not exist initially. However, we can infer the queuing delay D^1 of the second packet by Eq. (5) and use it as though it were the initial queuing delay, though we will lose the aggregated traffic captured in the first output gap. This phenomenon is regarded as a kind of noise that may slightly affect the "captured traffic ratio" determination (which will be discussed in the next section). However, the impact of this type of a noise on the accuracy of available bandwidth estimation may, in fact, be negligible when compared with the impact of probing noise.

After the queuing regions and queuing delays in a packet train have been determined, the so-called "captured traffic ratio (CTR)" can be quantified to determine the relationship between the probing rate and available bandwidth. Then, the next probing rate can be adjusted through binary search. In the next subsection, we describe how to calculate CTR.

2.2 Captured Traffic Ratio under Fluid Traffic Model

When a single packet pair is considered, the relationship between the probing rate (R) and available bandwidth ($Avbw$) can be expressed as

$$R > Avbw \text{ iff } \frac{s}{\Delta_{in}} > (TC - \frac{CT}{\Delta_{in}}), \quad (6)$$

where $R = \frac{s}{\Delta_{in}}$ denotes the probing rate of a packet pair, $\frac{CT}{\Delta_{in}}$ denotes the cross traffic rate captured by the packet pair, and $TC - \frac{CT}{\Delta_{in}}$ denotes $Avbw$. When a packet train is considered, Eq. (6) can be rewritten for all packet pairs of the j-th probing packet train with probing rate $R^j = \frac{s}{\Delta_{in}^j}$, which is composed of n packet pairs, as

$$R^j > Avbw \text{ iff } \sum_{i=1}^{n}(CT^i + s) > n \times TC \times \Delta_{in}^j, \quad (7)$$

where n denotes packet train's length, i.e., number of packet pairs in a packet train.

The cross traffic in Eq. (7) can be further classified into two types according to the kind of queuing region in which a packet pair operates. By distinguishing between different queuing regions, Eq. (7) can be rewritten as

$$R^j > Avbw \text{ iff}$$

$$\frac{\sum_{i=1}^{n}(CT^i+s)|_{QR^i \text{ is JQR}} + \sum_{i=1}^{n}(CT^i+s)|_{QR^i \text{ is DQR}}}{n \times TC \times \Delta_{in}^j} > 1. \quad (8)$$

However, no clue can be used to estimate the amount of cross traffic in a DQR. Consequently, only the cross traffic, which is captured in JQRs, can be exploited to determine the relationship between the probing rate and available bandwidth. Furthermore, under the case of the fluid traffic model, each packet pair in a packet train will operate in the same queuing region, i.e., JQR. Consequently, Eq. (8) can be rewritten as

$$R^j > Avbw, \text{ if } \frac{\sum_{i=1}^{n}(CT^i+s)|_{QR^i \text{ is JQR}}}{n \times TC \times \Delta_{in}^j} > 1. \quad (9)$$

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2006 proceedings.

Substituting $\Delta_{out}^{i} = \frac{CT^i + s}{TC}$ into Eq. (9), we have

$$R^j > Avbw, \quad \text{if} \quad \frac{\sum_{i=1}^{n} \Delta_{out}^{i}|_{QR^i \text{ is JQR}}}{n \times \Delta_{in}^{j}} = CTR > 1, \quad (10)$$

where CTR denotes the value of "captured traffic ratio (CTR)," which measures the total output gaps accumulated in JQRs per the total number of input gaps.

2.3 Probing Rate Adjustment

According to Eq. (10), the relationship between the probing rate of the j-th packet train and the available bandwidth can be determined. In order to get a more accurate estimate, binary search [6] is adopted to iteratively approximate the available bandwidth.

Given a probing rate R^j of the j-th packet train and an available bandwidth *Avbw*, the next probing rate and its boundaries (R_{min} and R_{max}) can be determined as

$$\begin{aligned} &R_{min} = R^j, \ \text{ if } R^j > Avbw, \\ &R_{max} = R^j, \ \text{ if } R^j < Avbw, \\ &R^{j+1} = (R_{min} + R_{max})/2, \end{aligned} \quad (11)$$

where R_{min} denotes the upper bound of $Avbw$ and R_{max} denotes the lower bound of $Avbw$. Although no guideline is available for setting the initial values of R_{min} and R_{max}, we recommend that R_{min} be set to the upper bound of the available bandwidth, which is equal to the capacity of the bottleneck link, and that R_{max} be set to the lower bound of the available bandwidth, which is, ideally, zero.

The above probing rate adjustment procedure stops if $|R_{min} - R_{max}| < \omega$ holds such that the degree of fluctuation of the available bandwidth is bound to ω, which is called the "estimation resolution." Under the fluid traffic model, ω can approximate zero at the expense of spending longer convergence time. When bursty cross traffic model is considered, we shall describe how to determine a suitable ω in Sec. 4. When the stop condition is satisfied, R_{max} is adopted as the estimated available bandwidth. In addition, R_{min} is adopted as the next initial probing rate when the available bandwidth procedure restarts.

So far, it can be seen that the proposed method tries to find the available bandwidth by pouring traffic into the network. Once the buffer of the router overflows, the incoming data will be dropped, leading to packet loss. Here, we assume that packet loss is caused by congestion, which occurs in a wired environment. Under this circumstance, if the receiver detects probing packet loss, then the sender has to adjust the new probing rate using the AIMD mechanism [4]. However, when the probing rate is reduced to be less than R_{max}, this adjustment is not helpful for enabling the probing rate to converge to the available bandwidth. Thus, we propose reducing the next probing rate to $\frac{R_{max}+R^j}{2}$ instead of $\frac{R^j}{2}$.

3 Proposed Method, JitterPath, Under Bursty Traffic Model

When bursty cross traffic is encountered, the output probing gaps, even carry useful information in measuring the cross traffic rate, are contaminated with probing noises. Liu *et al.* [10] first pointed out that the probing noise is caused by the superposition of probing and cross traffics if the probing rate is less than the available bandwidth. Specifically, given that $\lambda_{\Delta_{in}^{j}}^{i}$ being the average cross traffic rate during the inter-arrival time of the i-th packet pair in the bottleneck, s being the probe packet size, D^i being the queuing delay accumulated before the i-th probing packet, and $\tilde{I}_{\Delta_{in}^{j}}^{i}$ being the probing noise, the following relationship can be established [10]:

$$\Delta_{out}^{i} = \frac{\lambda_{\Delta_{in}^{j}}^{i} \times \Delta_{in}^{j}}{TC} + \frac{s}{TC} + \tilde{I}_{\Delta_{in}^{j}}^{i}. \quad (12)$$

The probing noise, $\tilde{I}_{\Delta_{in}^{j}}^{i}$, is the key leading to inaccurately measure cross traffic rate for PGM-based methods by means of exploiting the relationship between the output gap and input gap.

Since PGM methods care about the case when $\Delta_{out}^{i} = \Delta_{in}^{j}$ holds to infer the available bandwidth, Eq. (12) can be rewritten as

$$TC - (\lambda_{\Delta_{in}^{j}}^{i} + R^j) = \frac{\tilde{I}_{\Delta_{in}^{j}}^{i} \times TC}{\Delta_{in}^{j}}. \quad (13)$$

In Eq. (13), if probing rate R^j is equal to the available bandwidth and the cross traffic exactly fill the gap within the packet pair PP^i (the queuing region of PP^i is a JQR), then the probing noise $\tilde{I}_{\Delta_{in}^{j}}^{i}$ must be zero. In addition, if the cross traffic does not completely fill the gap within PP^i but creates a fake phenomenon that the $\Delta_{out}^{i} = \Delta_{in}^{j}$ (the queuing region of PP^i is a DQR), then probing noise exists. Under these circumstances, we define the rules of queuing region determination under the bursty cross traffic model as

$$\begin{aligned} &QR^i \text{ is } JQR, \ \text{ if } \tilde{I}_{\Delta_{in}^{j}}^{i} = 0; \\ &QR^i \text{ is } DQR, \ \text{ if } 0 < \tilde{I}_{\Delta_{in}^{j}}^{i} \le \Delta_{in}^{j}. \end{aligned} \quad (14)$$

In order to reduce the impact of probing noise on queuing region determination, in the next subsection, we discuss how to alleviate the probing noise by means of our queuing delay propagation model.

3.1 How to Alleviate Probing Noise?

In [10], Liu *et al.* investigated to quantify probing noise, $\tilde{I}_{\Delta_{in}^{j}}^{i}$, which is determined by the injected probing pack-

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2006 proceedings.

ets and accumulated queuing delay as:

$$\tilde{I}^i_{\Delta^j_{in}} = max(0, I^i_{\Delta^j_{in}} - \frac{s}{TC} - D^{i-1}), \quad (15)$$

where $I^i_{\Delta^j_{in}}$ is the idle period without serving any probing packet, and $0 \leq I^i_{\Delta^j_{in}} \leq \Delta^j_{in}$. It should be noted that $\tilde{I}^i_{\Delta^j_{in}}$ can only be estimated when probing rate is larger than available bandwidth. In the following, we analyze from two cases to show that the probing noise can be alleviated by means of our queuing region determination strategy. It should be noted that both cases discussed below consider the interference of probing noises so that a queuing region is erroneously determined to be a JQR but it is a DQR actually.

In the first case, we consider $\Delta OWD^i < 0$. In addition, we assume that $\Delta OWD^i + D^{i-1} > 0$ holds so that the queuing region of PP^i is determined to be a JQR based on Eq. (4). Based on Eq. (5), we can obtain $D^i < D^{i-1}$. If probing noises continue to exist, then a sequence of decreasing queuing delay can be built. When the decreasing queuing delay sequence converges to zero, i.e., $D^k = 0$, our method will accurately determine that the packet pair PP^{k+1} is a DQR based on Eq. (4) since $\Delta OWD^{k+1} + D^k < 0$. Therefore, the negative impact of probing noise will finally be stopped.

In the second case, we consider $\Delta OWD^i > 0$. Again, we assume that $\Delta OWD^i + D^{i-1} > 0$ holds so that the queuing region of PP^i is determined to be a JQR. Based on Eq. (5), we can obtain $D^i > D^{i-1}$. If probing noises continue to exist, then a sequence of increasing queuing delay can be built. Under this situation, we can see from Eq. (15) that the probing noise $\tilde{I}^k_{\Delta^j_{in}}$ is possibly to be eliminated when the queuing delay D^{k-1} is larger enough to make $I^k_{\Delta^j_{in}} < \frac{s}{TC} + D^{k-1}$.

Although we have proposed a strategy to stop or eliminate probing noises, they may still survive if the length of a probing packet train is not longer enough. Under this circumstance, the output gaps contaminate probing noises such that CTR in Eq. (10) is erroneously estimated. In the following subsection, we investigate how captured traffic ratio is affected by probing noise, and what the relationship between probing rate and available bandwidth is under bursty cross traffic model.

3.2 Captured Traffic Ratio under Bursty Cross Traffic

By taking summation for both sides of Eq. (12), we can derive

$$\frac{\sum_{i=1}^{n} \Delta^i_{out}}{n \times \Delta^j_{in}} = \frac{\sum_{i=1}^{n} \lambda^i_{\Delta^j_{in}} + n \times R^j}{n \times TC} + \frac{\sum_{i=1}^{n} \tilde{I}^i_{\Delta^j_{in}}}{n \times \Delta^j_{in}}, \quad (16)$$

for all packet pairs in the jth probing packet train, where the first term (let $\frac{\sum_{i=1}^{n} \lambda^i_{\Delta^j_{in}} + n\times R^j}{n\times TC} = \kappa$ hereafter) on the right side of Eq. (16) denotes the average cross traffic and probing rate, and the second term (let $\frac{\sum_{i=1}^{n} \tilde{I}^i_{\Delta^j_{in}}}{n\times \Delta^j_{in}} = \epsilon$ hereafter) denotes the average probing noise. If the average cross traffic and probing rate is larger than the tight-link capacity (TC), then the probing rate R^j is larger than the available bandwidth $Avbw$, i.e.,

$$\begin{array}{ll} R^j > Avbw, & \text{if } \kappa > 1, \\ R^j < Avbw, & \text{if } \kappa \leq 1. \end{array} \quad (17)$$

According to the definition of CTR shown in Eq. (10), it measures the ratio of total output gaps in JQRs to total input gaps. Since aggregated traffic in DQR gaps are not estimated, $\kappa \geq CTR$ holds. In [2], Hu *et al.* observed that available bandwidth observed for a long time in the Internet is stable and probing noise, which is caused by the fluctuation of bursty cross traffic, is also stable. Based on these, it is reasonable to conjecture that the probing noise ratio can be unchanged when the probing rate is less than the available bandwidth. As we known, κ is equal to CTR under fluid cross traffic model. When cross traffic is bursty, $\kappa - CTR$ should be bounded within ϵ, i.e., $|\kappa - CTR| < \epsilon$. Therefore, if $CTR < 1 - \epsilon$, then $\kappa \leq 1$ and probing rate is judged to be less than available bandwidth according to Eq. (17). In addition, if $CTR > 1$, then $\kappa > 1$ is guaranteed and probing rate is larger than available bandwidth. On the other hand, if $1 - \epsilon \leq CTR < 1$, the probing rate may be larger than the available bandwidth actually but their relationship may be erroneously estimated due to probing noise. Under this circumstance, the sender must use a higher probing rate to alleviate the effect of probing noises, as will be discussed in Sec. 4. The above discussions can be summarized as

$$\begin{array}{ll} R^j > Avbw, & \text{if } CTR > 1, \\ R^j < Avbw, & \text{if } CTR < 1 - \epsilon. \end{array} \quad (18)$$

Now, we describe how to estimate the probing noise ratio, ϵ. To simplify analysis, we assume the proposed method satisfies non-intrusiveness so that the probing packet size is close to zero and all queuing regions operate in JQRs (no undetermined traffic falling into DQR gaps exists). Therefore, CTR will be equal to κ. However, DQRs are erroneously estimated as JQRs when probing rate is smaller than available bandwidth. As a result, the probing noise ratio can only be determined when $CTR > 1$. By substituting $CTR = \kappa$ into Eq. (16), ϵ can be calculated as

$$\epsilon = \frac{\sum_{i=1}^{n} \Delta^i_{out}}{n \times \Delta^j_{in}} - CTR, \quad \text{if } CTR > 1. \quad (19)$$

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2006 proceedings.

4 Relationship between Probing Noise Ratio (ϵ) and Estimation Resolution (ω)

An effective way to alleviate probing noise is to pour more probing traffic into the network. This corresponds to probe longer packet trains, as described in Sec. 3.1. However, it should be careful to not pour too much amount of probing traffic to avoid unnecessary network collapse. In this section, we shall investigate the relationship between the probing noise ratio, ϵ, and the estimation resolution (also called the minimum compensated probing rate), ω. Since probing noise ratio, as described in Sec. 3.2, can be detected in practice, we propose to quantify the estimation resolution in the worse case where the probing rate is equal to the available bandwidth, which also implies that the bursty cross traffic with the largest rate is considered, so that our method is able to accommodate to bursty cross traffic with various rate. Due to the measurement time is finite, the probing noise may not be completely eliminated, as previously described in Sec. 3.1. As a result, the proposed method cannot correctly estimate the relationship between the probing rate and available bandwidth point if CTR falls within the interval $[1-\epsilon \quad 1)$. Under this circumstance, we need to probe with extra probing rate, ω, to eliminate probing noise.

Suppose that the average cross traffic rate is λ and the probing rate is equal to the available bandwidth, i.e., $R^j = TC - \lambda$. Therefore, the ratio of the total output gaps to total input gaps in Eq. (16) is equal to 1. In this study, the Pareto on/off cross traffic model is considered so that the maximum and minimum cross traffic rate can be represented as $\lambda+\tau_1$ and $\lambda-\tau_2$, respectively. In order to simplify analysis, $\tau_1 = \tau_2 = \tau$ is assumed. Under these circumstances, if a packet pair captures the cross traffic with a rate of $\lambda + \tau$, then the probing rate is larger than the available bandwidth point during the inter-arrival time of this packet pair in the bottleneck and its queuing region operates in JQR. On the other hand, the queuing region of a packet pair, which captures cross traffic with a rate of $\lambda - \tau$, may operate in DQR or JQR. If this packet pair can receive the propagated queuing delay of the preceding probing packets, this queuing delay is useful for this packet pair to operate in JQR, as revealed in Eqs. (14) and (15).

In the situation where the probing rate is equal to the available bandwidth, it is reasonable to assume that the number of packet pairs, which capture cross traffics with two different rates, are the same and are set to m. In addition, let k ($\leq m$) denote the number of packet pairs that capture cross traffic with a rate of $\lambda - \tau$ and receive sufficient accumulated queuing delays such that their queuing regions operate in JQR. Under these circumstances, the CTR of a packet train will be the amount of the aggregated traffic in these $m + k$ packet pairs divided by the maximum traffic that the bottleneck can process. The amount of cross traffic, $(\lambda + \tau) \times \Delta^j_{in}$, captured by the m packet pairs and the amount of cross traffic, $(\lambda - \tau) \times \Delta^j_{in}$, captured by the k packets are, respectively, used to substitute CT^i of Eq. (8), we can get

$$CTR = \frac{(m+k)\times TC + (m-k)\times\tau}{2m\times TC}, \qquad (20)$$

$$\epsilon = 1 - CTR = \frac{(m-k)\times(TC-\tau)}{2m\times TC}. \qquad (21)$$

In Eq. (21), the maximum value of ϵ (which can result in the maximum number of queuing regions determined to be DQRs) is found if $k = 0$. Substituting $k = 0$ into Eq. (21), we can derive ϵ as

$$\epsilon = \frac{TC-\tau}{2\times TC}. \qquad (22)$$

Since only the information carried in JQRs is helpful in estimating available bandwidth, the information carried in DQRs cannot be measured and, thus, wasted. In order to reduce the effect of probing noise, the sender needs to use a higher probing rate, i.e., the additional rate ω, to compensate the undermined aggregated traffic, which is composed of probing and cross traffic in those $m - k$ packet pairs operating in DQRs. This implies that the goal is to exhaust the unused bandwidth, such that the aggregated traffic will fill the gaps between packet pairs. To this end, the amount of undetermined aggregated traffic is first calculated as $(m-k)\times(R^j+\lambda-\tau)\times\Delta^j_{in}$. Then, we further consider that the probing rate is equal to the available bandwidth, i.e., $R^j = TC-\lambda$ holds with the aim that the bursty cross traffic with the largest rate is considered. Thus, the undetermined aggregated traffic is $(m-k)\times(TC-\tau)\times\Delta^j_{in}$. To satisfy the maximum compensation, $k = 0$ is set to obtain the maximum undetermined aggregated traffic $m\times(R^j+\lambda-\tau)\times\Delta^j_{in}$. In addition, the amount of additionally probed traffic is $2m\times\omega\times\Delta^j_{in}$ by using ω to replace TC. To enable effective compensation, the following condition must be guaranteed

$$2m\times\omega\times\Delta^j_{in} \geq m\times(TC-\tau)\times\Delta^j_{in}, \text{ i.e., } \omega \geq \frac{TC-\tau}{2}. \qquad (23)$$

Given Eqs. (22) and (23), the relationship between ω and ϵ can be derived as

$$\omega \geq TC\times\epsilon, \qquad (24)$$

which implies that the estimation resolution, ω, plays a key role in the trade-offs among the estimation accuracy, convergence speed, and intrusiveness. More specifically, it can be found that a small value of ω can yield more accurate estimation results at the expense of increased probing time (i.e., intrusiveness).

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2006 proceedings.

5 Simulation and Real-Network Results

In order to demonstrate the performance of our method, several simulations using ns2 [12] were conducted based on two different network models. Our method was also compared with PathChirp [16] and IGI [1]* in terms of the accuracy of available bandwidth estimation. PathChirp and IGI were selected because they are, respectively, typical examples of PRM- and PGM-based mechanisms. In this study, three different types of cross traffic, constant bit-rate (CBR) traffic, FTP traffic, and Poisson traffic were used for performance evaluation. CBR traffic was considered because it is the least bursty cross traffic, which approximates fluid traffic model. The cross traffic with a Poisson distribution was adopted to emulate the bursty and memoryless traffic over the Internet. The default parameters used in ns2 were set for cross traffics except that the packet size for the cross traffic was fixed at 1000 bytes.

In our simulations, the value of ω was initially set to 200Kbps, and its range could be dynamically changed according to Eq. (24). The lower bound of ω was adopted as the estimation resolution. The probing rate was initially set to half of the bottleneck link's capacity, and its variation during available bandwidth measurement is plotted using dash-dot curves in the figures shown below. For IGI, we adopted a packet train composed of 60 packet pairs and each packet size was 700 bytes, as suggested in [1] so that the best results could be obtained. For pathchirp, we adopted a packet train with a packet size of 700 bytes. With our method, the setting is the same as IGI for fair comparisons.

In addition to the extensive simulations, one real-network measurement is also performed to further confirm the achievable performance of our method.

5.1 Single-Hop Network Environment

The first network topology shown in Fig. 1 specifies a single-hop model, where Ps and Pr denote, respectively, the sender and receiver in the end-to-end probing path, and Cs and Cr denote, respectively, the sender and receiver in the cross traffic transmission path. Three different kinds of cross traffic were, respectively, used for evaluation of available bandwidth estimation. In the first scenario, the cross traffic contains the CBR traffic only, which implies that the cross traffic rate is stable. In the second scenario, the cross traffic is composed of FTP traffic only. In this case, we added the first FTP flow at the beginning of network transmission and added a new FTP flow from Cs to Cr per 25 seconds. For the third scenario, the cross traffic is a Poisson distribution with a rate of 8Mps, the bursty period is 5ms, and the idle period is 10ms.

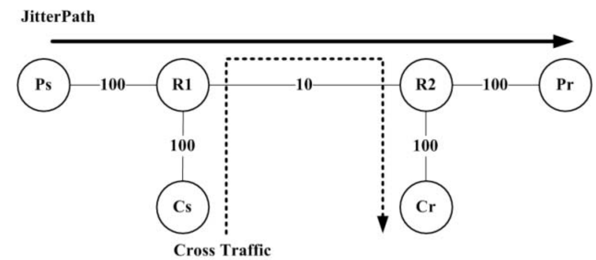

Figure 1: **A single-hop model, where the bottleneck link is the same as the tight-link, and bottleneck bandwidth is set to 10Mbps. The dash line denotes the path of cross traffic.**

The numerical results of available bandwidth estimation obtained using our method, IGI, and PathChirp, with respect to the three scenarios are depicted in Table. 1 for comparisons. We can observe from this table that our method almost achieves the measurement with the minimum errors. When the cross traffic is FTP or Poisson, the bursty behavior makes IGI and PathChirp erroneously estimate the relationship between probing rate and available bandwidth. However, since our method can properly alleviate the probing noise caused by bursty cross traffic, available bandwidth can be more accurately estimated. On the other hand, when the cross traffic is composed of CBR traffic only, the cross traffic approximates the fluid model, which fits the assumption of IGI, so that IGI can obtain the most accurate estimation. As for our method, due to the load of cross traffic is less than the default resolution (200Kbps), the probing noise cannot be completely stopped or eliminated after a short period of active probing such that larger estimation errors, when compared to IGI, are yielded.

Table 1: **Numerical Comparisons among JitterPath, IGI, and PathChirp in terms of mean-square error (MSE) calculated between the true and estimated available bandwidth. MSE is shown in Mbps.**

Methods	JitterPath	IGI	PathChirp
CBR	0.557244	0.111522	1.395860
FTP	0.566302	1.044908	0.949150
Poisson	0.963399	1.833687	1.398199

*In order to estimate available bandwidth in a multi-hop environment, the authors of [1] also proposed a PRM-based method, PTR. PTR uses a threshold-based strategy to detect the state where the probing rate is equal to the available bandwidth. However, the accuracy is closely related to the threshold. As shown in Eq. (13), we note that this threshold tends to be affected under bursty cross traffic model. Therefore, PTR is prone to underestimate the available bandwidth.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2006 proceedings.

5.2 Multi-hop Network Environment

In this section, two different multi-hop models were used to evaluate and compare our method, IGI, and PathChirp.

5.2.1 One-Hop Persistent

Fig. 2 shows a one-hop persistent model whose bottleneck can be shifted among all the links. In the network setting, the bandwidth of the first link (R1-R2) was set to 15 Mbps, the bandwidth of the second link (R2-R3) was set to 10 Mbps, and the bandwidth of the third link (R3-R4) was set to 15 Mbps. It should be noted that the available bandwidth of this end-to-end path is the minimum un-used bandwidth among the three links. The bottleneck was initially located at the bottleneck link (R2-R3) and could be shifted to other links if the cross traffic rate in each link was changed. The initial cross traffic rate in each link was 3 Mbps.

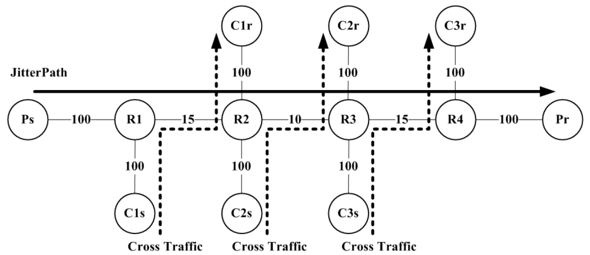

Figure 2: **One hop persistent of multi-hop network model. The bottleneck link may not be the tight-link.**

In this simulation, two bottleneck shifting scenarios were studied to verify the performance. First, the bottleneck was shifted to the link in front of the bottleneck link. This scenario could be achieved by increasing the cross traffic rate of link R1-R2 until link R1-R2 finally became the bottleneck. In the second case, the bottleneck was shifted from link R1-R2 to link R3-R4. This could be achieved by increasing the cross traffic rate of link R3-R4 until the bottleneck was finally transferred to R3-R4. In addition, it should be noted that the bottleneck was shifted from the second link to the first link at the 200th second, and was shifted from the first link to the third link at the 350th second.

When the cross traffic was composed of CBR flows only, our method was able to approximate the actual average available bandwidth with slight under-estimations, which are caused by the post narrow effect in that the packet pairs are queued again after the bottleneck. In fact, this phenomenon appears in the multi-hop environment. PathChirp yielded rather inaccurate estimations no matter whether the cross traffic rate was high or low. Meanwhile, we find that IGI only estimated the available bandwidth correctly when the bottleneck was exactly located at the bottleneck link. In summary, the major weakness of both PathChirp and IGI revealed in this simulation is that they tend to erroneously detect OWD increasing trend or obtain an erroneous queuing region.

When the cross traffic was composed of both CBR and FTP traffic, the bursty behavior of TCP flows could be simulated by first adding a FTP flow at the beginning of network transmission and adding a second FTP flow at the 200th second. It can be found that both IGI and PathChirp yielded inaccurate and oscillatory estimations. Compared with PathChirp and IGI, our method can obtain more stable estimations that are closer to the actual available bandwidth.

When the cross traffic was composed of both CBR and Poisson traffic, the Poisson traffic rate is 3Mbps, the bursty period is 5ms, and the idle period is 10ms. The bursty behavior of Poisson traffic could be simulated by adding three Poisson flows, which follow the same path as CBR flows. We use CBR flows to control the shift of tight-link. It can be found that both IGI and PathChirp yielded inaccurate and oscillatory estimations. Compared with PathChirp and IGI, our method can still obtain more stable estimations.

The numerical results are depicted in Table. 2 to summarize the above comparisons. We can observe from this table that our method consistently achieves the measurement with the minimum errors.

Table 2: **Comparisons among JitterPath, IGI, and PathChirp in one-hop persistent environment in terms of MSE, which is measured in Mbps.**

Methods	JitterPath	IGI	PathChirp
CBR	0.400793	1.158282	1.963439
CBR+FTP	0.646361	1.188017	1.613793
CBR+Poisson	0.607671	1.017480	1.259463

5.2.2 Path Persistent

Fig. 3 shows the path persistent model whose network setting is the same as Fig. 2 except that the bottleneck still locates in the bottleneck link during measurement. Again, the two scenarios and the three set of cross traffic used in the one-hop persistent environment were adopted for performance evaluation.

Again, we can observe from the results of available bandwidth estimation obtained from three different sets of cross traffic that similar conclusions can be drawn to be the same as in the one-hop persistent model. That is, both IGI and PathChirp basically yielded inaccurate and oscillatory estimations while JitterPath can still obtain more stable estimations. The numerical results are

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2006 proceedings.

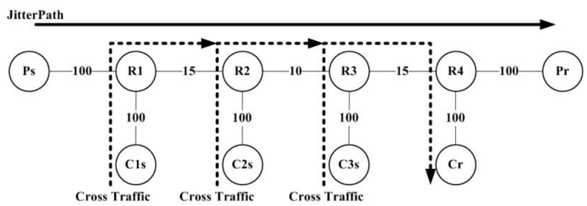

Figure 3: **Path persistent of multi-hop network model. Bottleneck locates in the bottleneck link during the measurement.**

depicted in Table. 3. Compared with the results obtained in the one-hop persistent model, it can be observed that smaller MSEs can be obtained in the path persistent model. This is because the bottleneck remains staying in the bottleneck link, which is similar to the single-hop environment.

Table 3: **Comparisons among JitterPath, IGI, and PathChirp in path persistent model in terms of MSE.**

Methods	JitterPath	IGI	PathChirp
CBR	0.259212	0.792976	0.538153
CBR+FTP	0.495831	0.999678	0.596466
CBR+Poisson	0.379730	0.636806	0.792812

5.3 Measurement Accuracy vs. Packet Train's Length

We examined the relationship between the estimation accuracy of our method and the length of probing packet train. The single-hop model shown in Fig. 1 was used in this simulation. The obtained numerical results are depicted in Table. 4. It can be seen that probing longer packet trains are able to achieve estimations with smaller MSEs no matter what kind of cross traffic is encountered. Moreover, this is consistent with our claim described in Sec. 3.1 that the probing noise can be more efficiently alleviated when a longer packet train is probed.

Table 4: **Measurement accuracy of JitterPath vs. packet train's length in a single-hop environment in terms of MSE.**

Train's length	60	100	150	200
CBR	0.5572	0.2868	0.2956	0.2461
FTP	0.5663	0.4615	0.2360	0.2048
Poisson	0.9634	0.7934	0.7276	0.6323

5.4 Real-Network Environment

Real-network measurement is also conducted to evaluate JitterPath and IGI†. It is known that the most accurate metric can be acquired by MRTG [14] report, however, MRTG need to access all links along the path and the knowledge of capacity of all links. As a result of these difficulties, we choose relative measurement error, which is also employed in [1], to measure the accuracy of these two approaches. The relative measurement error is defined as

$$\frac{|Avbw^e - throughput_{TCP}|}{BC}, \tag{25}$$

where $Avbw^e$ is the estimated available bandwidth, BC is the bottleneck link capacity, and $throughput_{TCP}$ is the bulk data transmission rate, which is measure by Iperf [19]. The two probing ends in the real-network environment are located at Academia Sinica and National Central University (Taiwan, ROC). The last mile (fast Ethernet) is the bottleneck link and its capacity is 100Mbps. Due to environmental constraint, we collected measurements from only one Internet path. However, these measurements were collected for at least 4 hours. The parameters setting for these three approaches here is the same as in simulations. The probing packet is of size 700 byte and the probing train is of length 60 packet pairs. Both available bandwidth and bulk TCP throughput are estimated for each time unit of 10 minutes.

The experiment result is shown in Fig. 4, which was produced by averaging the results obtained from pm 4:00 pm 8:00 in a week. We can observe from Fig. 4

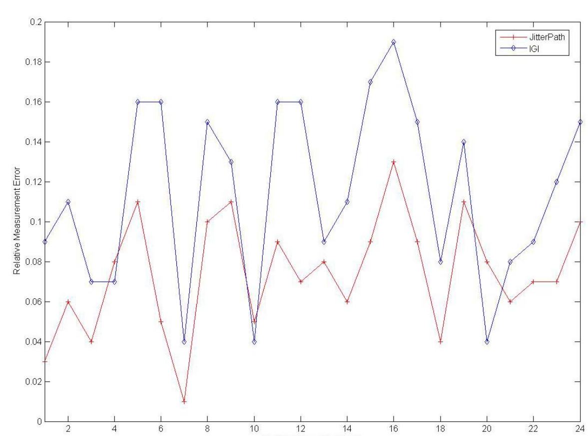

Figure 4: **Comparison of real-network measurement between JitterPath and IGI.**

that the relative measurement errors obtained from our

† Since IGI and pathchirp obtained comparable results in our simulations, we only selected IGI for comparison in real-network experiment.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2006 proceedings.

method (JitterPath) is mostly less than those obtained from IGI and the differences are significant. As a whole, the real-network result is consistent with the simulation results described previously.

6 Conclusion

Traditional transport protocols are unable to provide stable transmission due to ignorance of the available bandwidth. Meanwhile, end-to-end available bandwidth estimation has been found to be helpful for congestion control of multimedia transmission. In this paper, we have proposed a reliable available bandwidth estimation scheme, JitterPath, which is based on the one-way delay jitter and queuing region propagation. The key to accurate bandwidth estimation is to exploit the relationships between accumulated queuing delays and one-way delay jitter so that the attribute of the queuing region in each packet pair can be determined. Then, the captured traffic ratio, which is defined as the ratio of the total number of output gaps accumulated in JQRs to the total number of input gaps, can be used to specify the relationship between the probing rate and available bandwidth. A binary search-based probing rate adjustment mechanism has been proposed to approximate the available bandwidth with an error that is within the estimation resolution, which has been proved to be closely related to the probing noise ratio. Our method has been, respectively, presented for both the fluid and bursty traffic models.

In this study, we have investigated to find that probing noises can be finally stopped or eliminated after a sufficient number of packets have been probed. In order to quickly reduce the impact of probing noises, it is intuitive that the size of the input gap should be small. However, the size of the probing packets has also to be small in order to maintain the probing rate unchanged. Under these circumstances, a shorter packet train should be used for fast estimation at the expense of providing a short-term but unreliable measurement. To get a long-term and reliable measurement, a longer probing train is required. This tradeoff deserves further researches.

References

[1] N. Hu and P. Steenkiste, "Evaluation and Characterization of Available Bandwidth Techniques," *IEEE Journal on Selected Areas in Communication: Special Issue in Internet and WWW Measurement, Mapping, and Modeling*, 2003.

[2] N. Hu, L. Li, Z. M. Mao, P. Steenkiste and J. Wang, "A Measurement Study of Internet Bottlenecks," *Proceedings of the 24th IEEE Infocom*, March. 2005.

[3] Y. C. Huang, C. S. Lu, and H. K. Wu, "Reliable Available Bandwidth Estimation Based on Distinguishing Queuing Regions and Resolving False Estimations," *Proc. IEEE Global Telecommunications Conference (Globecom),* Dallas Texas, USA, 2004.

[4] V. Jacobson, "Congestion Avoidance and Control," *Proceedings of ACM SIGCOMM*, pp.314-329, September 1988.

[5] M. Jain and C. Dovrolis, "End-to-End Available Bandwidth: Measurement Methodology, Dynamic, and Relation with TCP Throughput," *Proceedings of ACM SIGCOMM 2002*, 2002.

[6] M. Jain and C. Dovrolis, "Pathload: A Measurement Tool for End-to-End Available Bandwidth," *Proc. Passive and Active Measurements*, March 2002.

[7] M. Jain and C. Dovrolis, "Ten Fallacies and Pitfalls on End-to-End Available bandwidth Estimation," *Proceedings of the conference on Internet measurement conference*, October 2004.

[8] S. Keshav, "A Control-Theoretic Approach to Flow Control," *Proc. ACM SIGCOMM*, pp. 3-15, 1991.

[9] Q. Liu, and J. Hwang, "End-to-End Available Bandwidth Estimation and Time Measurement Adjustment for Multimedia QoS," *Proc. IEEE Int. Conf. on Multimedia and Expo*, 2003.

[10] X. Liu, K. Ravindran, B. Liu, and D. Loguinov, "Single-Hop Asymptotics in Available Bandwidth Estimation: Sample-Path Analysis," *ACM SIGCOMM Internet Measurement Conference*, October. 2004.

[11] B. Melander, M. Bjorkman, and P. Gunningberg, "A New End-to-End Probing and Analysis Method for Estimating Bandwidth Bottlenecks," *Proc. Global Internet Symposium*, 2000.

[12] ns2 [Online]. http://www.isi.edu/nsnam/ns.

[13] J. Navratil, and R. L. Cottrell, "ABwE: A Practical Approach to Available Bandwidth Estimation," *Proc. Passive and Active Measurement Workshop*, 2003.

[14] T. Oetiker and D. Rand. Multi Router Traffic Grapher. http://people.ee.ethz.ch/ oetiker/webtools/mrtg/.

[15] V. J. Ribeiro, M. Coates, R. H. Riedi, S. Sarvotham, and R. G. Baraniuk, "Multifractal Cross Traffic Estimation," *Proc. of ITC specialist seminar on IP traffic Measurement*, 2000.

[16] V. J. Ribeiro, R. H. Riedi, R. G. Baraniuk, J. Navratil, and L. Cottrell, "PathChirp: Efficient Available Bandwidth Estimation for Network Paths," *Proc. Passive and Active Measurement Workshop*, 2003.

[17] V. J. Ribeiro, R. H. Riedi and R. G. Baraniuk, "Locating Available Bandwidth Bottlenecks," *Second International Workshop on Protocols for Fast Long-Distance Networks*, October 2004.

[18] J. Strauss, D. Katabi, and F. Kaashoek, "A Measurement Study of Available Bandwidth Estimation Tools," *Proceedings of the conference on Internet measurement Workshop*, pp. 27-29, 2003.

[19] A. Tirumala and J. Ferguson. Iperf. [Online]. Available: http://dast.nlanr.net/Projects/Iperf

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2006 proceedings.