The Probe Gap Model can Underestimate the Available Bandwidth of Multihop Paths

Li Lao
Google Inc.
Santa Monica, CA
llao@google.com

Constantine Dovrolis
Georgia Tech
Atlanta, GA
dovrolis@cc.gatech.edu

M.Y. Sanadidi
UCLA
Los Angeles, CA
medy@cs.ucla.edu

ABSTRACT

The Probe Gap Model (PGM) was proposed as a lightweight and fast available bandwidth estimation method. Measurement tools such as Delphi and Spruce are based on PGM. Compared to estimation methods that require multiple iterations with different probing rates, PGM uses a single probing rate and it infers the available bandwidth from a direct relation between the input and output rates of measurement packet pairs. An important assumption behind the PGM model is that the measured path has a single bottleneck link that determines the available bandwidth of the end-to-end path. In this letter, we show that, even though PGM is accurate in the case of a single queue, it cannot estimate the available bandwidth of multi-hop paths, even if there is a single bottleneck in the path. Whether PGM is accurate or not depends on the routing of cross traffic relative to the measurement traffic. PGM is accurate when the cross traffic follows the same path with the measurement traffic. In the general case, however, PGM can significantly underestimate the available bandwidth of an end-to-end path.

Categories and Subject Descriptors

C.2.3 [**Computer-Communication Networks**]: Network Operations—*Network Monitoring*

General Terms

Performance, Verification

Keywords

Network capacity, available bandwidth, packet pair dispersion, Probe Gap Model

1. INTRODUCTION

The problem of end-to-end available bandwidth estimation has received significant research interest [1, 2, 5, 6, 7, 8, 10]. The available bandwidth of a link is defined as the average residual (or spare) capacity of that link. The link with the minimum available bandwidth in a network path is called the *tight link* of the path, whereas the link with the minimum capacity is called the *narrow link*. The *end-to-end available bandwidth* of a network path is equal to the available bandwidth of the tight link.

Existing available bandwidth estimation techniques can be grouped into two classes [3]. The first is the *Probe Rate Model (PRM)*, also known as Iterative Probing. In PRM, packet pairs (or packet trains) are sent at different rates, while the receiver observes the relation between the input and output probing rates. When the output rate is lower than the input rate, the probing rate is higher than the available bandwidth. PRM tools can use this fact to "converge" into a range within which the available bandwidth varies, adjusting the input probing rate.

The second class is the *Probe Gap Model (PGM)*, also known as Direct Probing. In PGM, packet pairs (or packet trains) are sent to the path at a single rate. This probing rate is set to the capacity of the tight link, and so it is larger than (or equal to) the available bandwidth of the path. The main component of PGM is a mathematical relation between the input and output rates of a probing packet pair, under a fluid traffic model, based on which PGM tools such as *Delphi* [7] or *Spruce* [10] estimate the end-to-end available bandwidth. Even though that relation has been derived for a single queue model, it has been claimed that PGM is also accurate in multi-hop paths as long as there is a single bottleneck [10].

In this letter, we focus on the accuracy of the PGM method, and of Spruce in particular, in multi-hop paths. We show that in general, even if there is a single bottleneck in the path, *PGM tools provably underestimate the end-to-end available bandwidth*. Specifically, whether PGM is accurate or not depends on the routing of cross traffic relative to the measurement traffic. PGM is accurate when the cross traffic follows the same path with the measurement traffic. In the general case, however, PGM can significantly underestimate the available bandwidth of an end-to-end path.

The rest of this letter is organized as follows. In Section 2, we provide some relevant background. Then, in Section 3, we evaluate the accuracy of PGM mathematically, first at a single-hop path and then at a two-hop path. The analysis of the latter shows that PGM/Spruce are not accurate in the general case. Simulation results confirm the analysis in Section 4. We conclude in Section 5.

2. BACKGROUND

In PGM, it is assumed that the tight link is the same with the narrow link, referred simply as *bottleneck*. Further, PGM assumes that the capacity C of the bottleneck is known in advance. Spruce relies on probing packet pairs, but the PGM method could also be implemented with packet trains.[1] PGM is based on the relationship between the in-

---

[1] The duration of the packet trains (determined by the number and size of the probing packets in the train) corresponds to the averaging timescale over which the available bandwidth is defined. The variance of the available bandwidth

terarrival (also referred to as *dispersion*) of probing packets at the input and output of the measured path. Specifically, if the size of the probing packets is L and the packet pair is sent to the path at the rate of the bottleneck capacity C, the input dispersion of the packet pair is $\Delta_{in} = L/C$. According to PGM and Spruce, the available bandwidth A can be estimated from the measured dispersion Δ_{out} of the packet pair at the receiver, as follows:

$$A_{Spruce} = C \left(1 - \frac{\Delta_{out} - \Delta_{in}}{\Delta_{in}} \right) \qquad (1)$$

In the next section, we show that this is equal to the available bandwidth, under a fluid traffic model, of a single-queue. We also show, however, that A_{Spruce} can be less than the available bandwidth of a multi-hop path, even if there is a single bottleneck.

PRM also assumes a single tight link, but it does not presume knowledge of that link's capacity, and it does not assume that the narrow link is the same with the tight link. Instead, it relies on iterative probing at various rates based on the following simple observation. When the probing rate is higher than the available bandwidth, the tight link becomes saturated and a queue builds up. The receiver can detect that the tight link is saturated because the delays of successive probing packets increase, or equivalently, because the probing rate at the receiver is less than the probing rate at the sender. The receiver can then estimate the available bandwidth as the minimum probing rate that does not saturate the tight link. Tools such as TOPP [6], Pathload [2], PathChirp [8], IGI and PTR [1] are all based on variations of PRM. We clarify that the PRM approach does not suffer from the underestimation error we focus on in this letter. The reason is that PRM does not rely on any explicit relation between the input rate, available bandwidth, and output rate at the tight link.

In addition to the previous available bandwidth estimation techniques, there are also efforts to evaluate and compare such techniques in a systematic way with controllable and repeatable experiments [9]. Recently, Liu et al. presented a stochastic model to analyze the asymptotic behavior of packet pair/train probing over multi-hop network paths [5]. That paper also examined the estimation errors that can take place in multi-hop paths even with fluid traffic.

3. ANALYSIS OF PGM/SPRUCE

Following the standard assumptions behind the Probe Gap Model, we assume that that each link can be modeled as a First-Come-First-Served queue and that the cross traffic in the measured path follows the fluid model. This means that if the cross traffic at a link has rate R, then the amount of cross traffic that arrives in that link during *any interval* of length Δ is $R\Delta$. Further, as previously noted, PGM assumes that the tight link is the same with the narrow link, and that the capacity C of this single bottleneck link is known.

We denote the capacity, utilization, and available bandwidth of the i'th link as C_i, u_i, and A_i, respectively. We have that $u_i = (C_i - A_i)/C_i$, and so the utilization term

<hr>

decreases as the averaging timescale increases, and so longer trains cause a lower estimation error when the cross traffic is bursty [4]. That source of estimation error is different than the underestimation in multi-hop paths, even with fluid traffic, that we focus on in this letter.
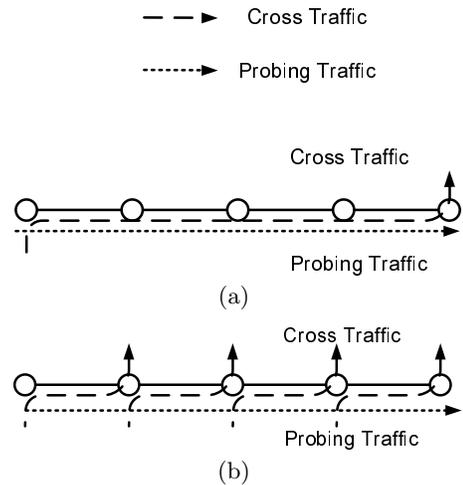
Figure 1: Path persistent (a') and one-hop persistent (b') cross traffic.

includes the cross traffic but not the measurement traffic. The end-to-end available bandwidth A is

$$A = \min_i A_i = \min_i C_i(1 - u_i) = C(1 - u) \qquad (2)$$

where u is the utilization of the bottleneck. The dispersion of the packet pair at the input of the first queue is Δ_{in}, while the dispersion of the packet pair at the output of the i-th link is Δ_i. Similarly, we denote the input rate of the packet pair as R_{in}, and the rate of the pair at the output of the i-th link as R_i. The size of the probing packets is L and so $R_i = L/\Delta_i$. The receiver can only measure the dispersion and rate at the output of the last link, denoted by Δ_{out} and R_{out}, respectively. In Spruce, $\Delta_{in} = L/C$, i.e., each packet pair is sent at the rate of the bottleneck's capacity. This guarantees that the input probing rate is higher than the available bandwidth of the path.

3.1 Single-hop path

First consider a single queue. The two probing packets arrive at the queue *back-to-back*, i.e., with dispersion $\Delta_{in} = L/C$. The amount of cross traffic that arrives at the queue between the two packets of the packet pair is $X = (C - A)\Delta_{in} = uL$, and so the dispersion of the packet pair at the output of the link is increased to:

$$\Delta_{out} = \frac{L + X}{C} = \Delta_{in}(1 + u) \qquad (3)$$

The Spruce receiver measures Δ_{out} and then estimates the available bandwidth from Eq.1 as:

$$A_{Spruce} = C \left(1 - \frac{\Delta_{in}(1 + u) - \Delta_{in}}{\Delta_{in}} \right) = A \qquad (4)$$

so, under the assumptions of the PGM model, Spruce can accurately estimate the available bandwidth of a single-hop path.

3.2 Multi-hop path

In the case of multi-hop paths, the cross traffic routing relative to the measured path is important. To see why, consider the two cases of Fig.1. *Path persistent traffic* follows

the same path with the measurement traffic. *One-hop persistent traffic* shares only a single queue with the measurement traffic. In the path persistent case, cross traffic only intervenes between the two probing packets at the first queue, and so the amount of traffic between the packet pair remains constant in the rest of the path.[2] In the one-hop persistent case, the amount of cross traffic that intervenes between the packet pair at hop i is equal to $(C_i - A_i)\Delta_{i-1}$ (with $\Delta_0 = \Delta_{in}$). In the following, we show that PGM/Spruce is accurate in the path persistent case, but not in the more general case of non-persistent traffic (e.g., one-hop persistent). Also, because the underestimation error we focus on takes place even in a two-hop path, we analyze that case for simplicity.

3.2.1 Path Persistent Cross Traffic

The cross traffic that intervenes between the packet pair is the traffic that arrives just after the arrival of the first probing packet and before the arrival of the second. So, the cross traffic between the packet pair at the first hop is $X = u_1 C_1 \Delta_{in}$, where $\Delta_{in} = L/C$. Due to the path persistent nature of the cross traffic, the arrival rate of the latter is the same at both the first and second queues, i.e., $u_1 C_1 = u_2 C_2$.

If the bottleneck is the first link ($C = C_1$), the dispersion at the output of the first link is:

$$\Delta_1 = \frac{L + X}{C_1} = \Delta_{in}(1 + u_1) \qquad (5)$$

The dispersion stays constant at the second link because $C_1 < C_2$. So, $\Delta_{out} = \Delta_{in}(1+u_1)$, which is the same relation we derived for the single queue. Thus, $A_{Spruce} = C_2(1 - u_2) = A$.

If the bottleneck is the second link ($C = C_2$), we need to consider the following two cases. First, if $(L+X)/C_1 < \Delta_{in}$, which is equivalent to $u_1 < 1 - C_2/C_1$, the first link can transmit the first probing packet and the intervening cross traffic before the second probing packet arrives. In this case the dispersion of the packet pair does not increase at the first link, i.e., $\Delta_1 = \Delta_{in}$. At the second link the dispersion is increased to:

$$\Delta_{out} = \frac{L + X}{C_2} = \frac{L + u_2 C_2 \Delta_{in}}{C_2} = \Delta_{in}(1 + u_2) \quad (6)$$

Thus, $A_{Spruce} = C_2(1 - u_2) = A$.

Otherwise, if $(L + X)/C_1 \geq \Delta_{in}$, which is equivalent to $u_1 \geq 1 - C_2/C_1$, the dispersion of the packet pair at the output of the first link is increased to:

$$\Delta_1 = \frac{L + X}{C_1} = \frac{L}{C_1} + u_1 \frac{L}{C_2} \qquad (7)$$

The dispersion at the output of the second link is further increased to:

$$\Delta_{out} = \Delta_1 \frac{C_1}{C_2} = \Delta_{in}(1 + u_2) \qquad (8)$$

Thus, as in the previous case, $A_{Spruce} = A$.

So, no matter if the bottleneck is the first or the second link, and for any utilization at the non-bottleneck link, Spruce can accurately estimate the available bandwidth of a two-hop path with path persistent cross traffic.

[2] Note that our definition of path persistent traffic is different than that of [5].

3.2.2 One-Hop Persistent Cross Traffic

With one-hop persistent cross traffic, the amount of cross traffic that arrives between the packet pair (i.e., just after the arrival of the first probing packet and before the arrival of the second) at hop i is $X_i = (C_i - A_i)\Delta_{i-1}$. It takes $(L + X_i)/C_i$ time units to transmit the first probing packet and this cross traffic. So, if $(L + X_i)/C_i \geq \Delta_{i-1}$, which is equivalent to $R_{i-1} \geq A_i$, the packet pair dispersion will be increased from $\Delta_{i-1} = L/R_{i-1}$ to:

$$\Delta_i = \frac{L + X_i}{C_i} = \frac{R_{i-1}\Delta_{i-1} + (C_i - A_i)\Delta_{i-1}}{C_i}$$

and so

$$\Delta_i = \Delta_{i-1}\left(1 + \frac{R_{i-1} - A_i}{C_i}\right) \quad , \quad R_{i-1} \geq A_i \qquad (9)$$

On the other hand, if $R_{i-1} < A_i$, link i can complete the transmission of the first probing packet and of X_i before the second probing packet arrives, and so the dispersion of the packet pair remains constant:

$$\Delta_i = \Delta_{i-1}, \quad R_{i-1} < A_i \qquad (10)$$

Returning to the two-hop model, we now distinguish two cases depending on whether the bottleneck is the first or the second link.

The bottleneck is the first link. In this case, we have that $R_{in} = L/\Delta_{in} = C_1 \geq A_1$, and so based on Eq.9,

$$\Delta_1 = \Delta_{in}(1 + \frac{R_{in} - A_1}{C_1}) = \Delta_{in}(1 + u_1), \qquad (11)$$

The important point, here, is that the probing rate $R_1 = L/\Delta_1$ after the first queue may be larger than the available bandwidth at the second queue. Specifically, it is easy to show that $R_1 \geq A_2$ if $C_1 \geq A_2(1+u_1)$. Because the first link is the bottleneck, however, we must also have that $C_1 < C_2$ and $A_1 < A_2$, or that $C_1 < \min\{C_2, A_2/(1 - u_1)\}$. So, $R_1 \geq A_2$ when the capacity of the bottleneck falls in the following range:

$$A_2(1 + u_1) \leq C_1 < \min\{C_2, A_2/(1 - u_1)\} \qquad (12)$$

When $R_1 \geq A_2$, the dispersion of the packet pair at the second queue increases to

$$\Delta_2 = \Delta_1(1 + \frac{R_1 - A_2}{C_2}) \qquad (13)$$

After some straightforward algebra, it can be shown that the Spruce estimate is:

$$A_{Spruce} = C_1(1 - \frac{\Delta_2 - \Delta_{in}}{\Delta_{in}}) = A - \frac{C_1}{C_2}[C_1 - A_2(1 + u_1)]$$

$$(14)$$

Note that the bias term is positive, and so Spruce underestimates the available bandwidth. Further, the bias can be significant. For example, suppose that C_1=8Mbps, A_1=2.4Mbps, C_2=10Mbps, and A_2=3Mbps. The bias in this example would be 2.32Mbps, and the available bandwidth according to Spruce would be A_{Spruce}=0.08Mbps.

On the other hand, when $R_1 < A_2$ the dispersion of the packet pair does not increase at the second queue, and so $\Delta_2 = \Delta_1$. In that case, $\Delta_{out} = \Delta_{in}(1+u_1)$ and so $A_{Spruce} = A$, i.e., Spruce is accurate.

The bottleneck is the second link. In this case, $R_{in} = L/\Delta_{in} = C_2 \geq A_2$. Here, we have that $R_{in} \geq A_1$ when the capacity of the bottleneck falls in the following range:

$$A_1 \leq C_2 < \min\{C_1, A_1/(1 - u_2)\} \qquad (15)$$

When $R_{in} \geq A_1$, the dispersion of the packet pair at the first queue increases to

$$\Delta_1 = \Delta_{in}(1 + \frac{R_{in} - A_1}{C_1}) \qquad (16)$$

The probing rate after the first queue is

$$R_1 = \frac{C_1 C_2}{C_1 + C_2 - A_1} \qquad (17)$$

It can be shown that $R_1 > A_2$ (because $C_1 > A_2$ and $A_1 > A_2$), and so the dispersion of the packet pair at the second queue increases further to

$$\Delta_2 = \Delta_1(1 + \frac{R_1 - A_2}{C_2}) \qquad (18)$$

After some algebra, it can be shown that the Spruce estimate is less than the available bandwidth:

$$A_{Spruce} = C_2 \quad 1 - \frac{\Delta_2 - \Delta_{in}}{\Delta_{in}} \quad \leq A \qquad (19)$$

The bias is a rather long expression in this case, and we do not include it here. It can also be significant, however. For example, suppose that C_1=10Mbps, A_1=5Mbps, C_2=8Mbps, and A_2=4Mbps. The bias in this example would be approximately 1.2Mbps and the available bandwidth according to Spruce would be $A_{Spruce} \approx$2.8Mbps.

On the other hand, when $R_{in} < A_1$ the dispersion of the packet pair does not increase at the first queue, and so $\Delta_1 = \Delta_{in}$. In that case, $\Delta_{out} = \Delta_1(1 + u_2)$, and so $A_{Spruce} = A$, i.e., Spruce is accurate.

3.2.3   Summary

To summarize the analysis of the two-hop path, Spruce is accurate when the cross traffic is path persistent. That is a very special case however. In the more general case where the two links carry different cross traffic, Spruce results in available bandwidth underestimation when the first link is the bottleneck and $A_2(1+u_1) \leq C_1 < \min\{C_2, A_2/(1-u_1)\}$, or when the second link is the bottleneck and $A_1 \leq C_2 < \min\{C_1, A_1/(1 - u_2)\}$. In both cases, the underestimation error can be significant.

4.   SIMULATION STUDY

In this section, we present simulation results of PGM probing in two paths with two-hops each. The link capacities are {6, 10}Mbps and {10, 6}Mbps, respectively. The cross traffic is constant-bit-rate (CBR) traffic with a small packet size of 40 bytes, similar to the fluid model. The probing packet size is 1500 bytes.

In the case of path persistent cross traffic, Fig.2 shows that PGM results in accurate estimation independent of whether the bottleneck is the first or the second link, and for the entire utilization range, as expected from the analysis of Section 3.2.1.

In the case of one-hop persistent traffic, we vary the utilization of the bottleneck link and of the non-bottleneck link, and plot both the real and the estimated available bandwidth according to PGM. Fig.3 shows the results for
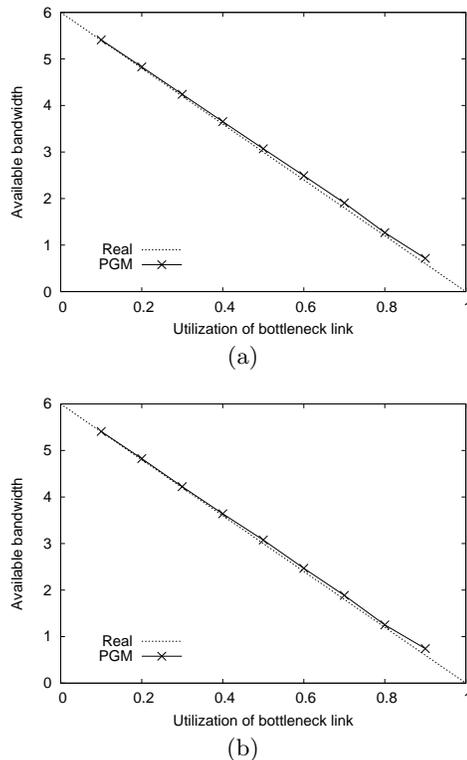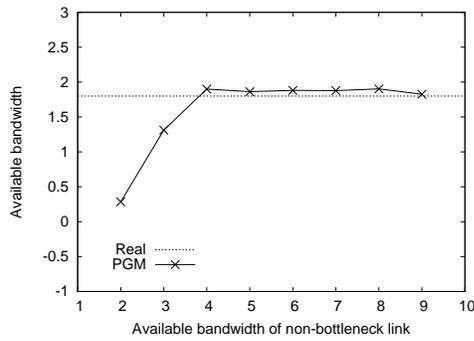

Figure 2: Estimated available bandwidth for a two-hop path with (a) C_1=6Mbps and C_2=10Mbps, and (b) C_1=10Mbps and C_2=6Mbps.   The cross traffic follows the path persistent model.
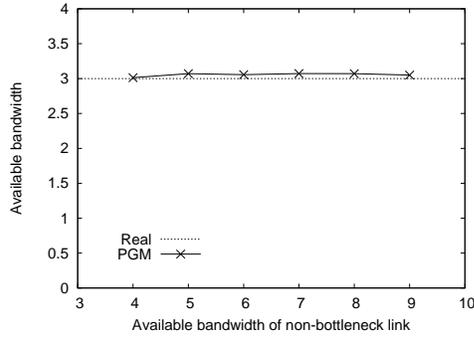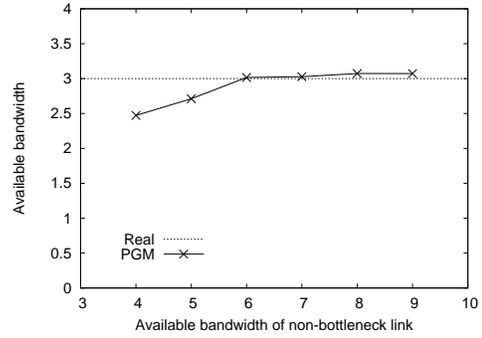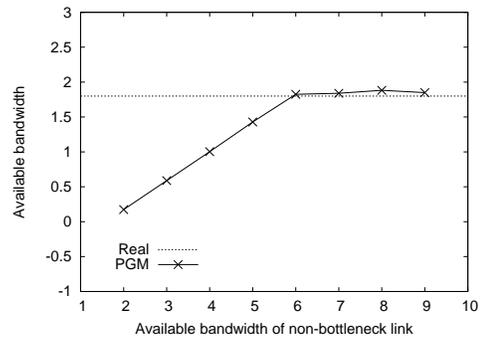
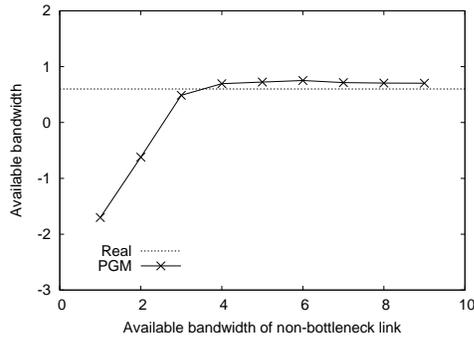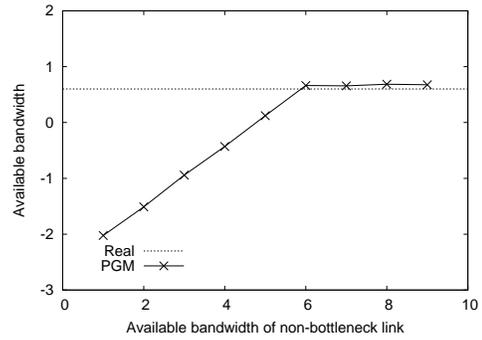**Figure 3: Estimated available bandwidth for a two-hop path with C_1=6Mbps and C_2=10Mbps. The available bandwidth A_1 of the bottleneck link is: (a) 3Mbps (b) 1.8Mbps and (c) 0.6Mbps. The cross traffic follows the one-hop persistent model.**

Figure 4: Estimated available bandwidth for a two-hop path with C_1=10Mbps and C_2=6Mbps. The available bandwidth A_2 of the bottleneck link is: (a) 3Mbps (b) 1.8Mbps and (c) 0.6Mbps. The cross traffic follows the one-hop persistent model.

the first path and Fig.4 for the second path. Notice that PGM underestimates the end-to-end available bandwidth when the available bandwidth of the non-bottleneck link is less than a certain threshold. This threshold is determined by the conditions derived in Section 3.2.2. Specifically, in Fig.3, underestimation occurs when the available bandwidth $A_2 = C_2(1 - u_2)$ at the non-bottleneck link is less than $C_1/(1 + u_1)$. In Fig.4, underestimation occurs when the available bandwidth $A_1 = C_1(1 - u_1)$ at the non-bottleneck link is less than the capacity C_2 of the bottleneck, i.e., when $A_1 <$6Mbps. In both cases, the estimation error increases as the available bandwidth of the non-bottleneck link approaches the available bandwidth of the bottleneck. As a matter of fact, the estimation error can be so large that the PGM estimate is sometimes negative! Spruce filters out packet pairs that result in negative estimates.

5. CONCLUSIONS

It would certainly be useful to have an available bandwidth estimation method that does not require iterative probing. Such a method would be faster and more lightweight than PRM. However, this letter showed that PGM cannot estimate the end-to-end available bandwidth of multi-hop paths in the general case, and so it cannot be viewed as an accurate replacement of PRM. Finally, we note that available bandwidth measurements are also subject to other errors, due to the effects of traffic burstiness (real traffic is certainly not a fluid), or due to more practical issues such as inaccurate timestamps. For a discussion of those issues we also refer the reader to [3].

Acknowledgements

6. REFERENCES

[1] N. Hu and P. Steenkiste. Evaluation and characterization of available bandwidth probing techniques. *IEEE Journal on Selected Areas in Communications*, 21(6):879–894, Aug. 2003.

[2] M. Jain and C. Dovrolis. End-to-end available bandwidth: Measurement methodology, dynamics, and relation with TCP throughput. *IEEE/ACM Transactions on Networking*, 11(4):537–549, Aug. 2003.

[3] M. Jain and C. Dovrolis. Ten fallacies and pitfalls on end-to-end available bandwidth estimation. In *Proceedings of Internet Measurement Conference (IMC'03)*, Oct. 2003.

[4] X. Liu, K. Ravindran, B. Liu, and D. Loguinov. Single-Hop Probing Asymptotics in Available Bandwidth Estimation: A Sample-Path Analysis. In *Proceedings of Internet Measurement Conference (IMC'04)*, 2004.

[5] X. Liu, K. Ravindran, and D. Loguinov. Multi-hop probing asymptotics in available bandwidth estimation: Stochastic analysis. In *Proceedings of Internet Measurement Conference (IMC'05)*, Oct. 2005.

[6] B. Melander, M. Bjorkman, and P. Gunningberg. A new end-to-end probing and analysis method for estimating bandwidth bottlenecks. In *Proceedings of IEEE Global Internet Symposium*, 2000.

[7] V. Ribeiro, M. Coates, R. Riedi, S. Sarvotham, B. Hendricks, and R. Baraniuk. Multifractal cross-traffic estimation. In *Proceedings of ITC Specialist Seminar On IP Traffic Measurement, Modeling, and Management*, Sept. 2000.

[8] V. Ribeiro, R. Riedi, R. Baraniuk, J. Navratil, and L. Cotterll. pathchirp: Efficient available bandwidth estimation for network paths. In *Proceedings of Passive and Active Measurements (PAM) Workshop*, Apr. 2003.

[9] A. Shriram, M. Murray, Y. Hyun, N. Brownlee, A. Broido, and M. Fomenkov. Comparison of Public End-to-end Bandwidth Estimation Tools on High-Speed Links. In *Proceedings Passive and Active Measurements (PAM)*, 2005.

[10] J. Strauss, D. Katabi, and F. Kaashoek. A measurement study of available bandwidth estimation tools. In *Proceedings of Internet Measurement Conference (IMC'03)*, Oct. 2003.