What do packet dispersion techniques measure?

Constantinos Dovrolis
University of Wisconsin
dovrolis@ece.wisc.edu

Parameswaran Ramanathan
University of Wisconsin
parmesh@ece.wisc.edu

David Moore
CAIDA
dmoore@caida.org

Abstract— **The packet pair technique estimates the capacity of a path (bottleneck bandwidth) from the dispersion (spacing) experienced by two back-to-back packets [1][2][3]. We demonstrate that the dispersion of packet pairs in loaded paths follows a multimodal distribution, and discuss the queueing effects that cause the multiple modes. We show that the path capacity is often not the global mode, and so it cannot be estimated using standard statistical procedures. The effect of the size of the probing packets is also investigated, showing that the conventional wisdom of using maximum sized packet pairs is not optimal. We then study the dispersion of long packet trains. Increasing the length of the packet train reduces the measurement variance, but the estimates converge to a value, referred to as Asymptotic Dispersion Rate (ADR), that is lower than the capacity. We derive the effect of the cross traffic in the dispersion of long packet trains, showing that the ADR is not the available bandwidth in the path, as was assumed in previous work. Putting all the pieces together, we present a capacity estimation methodology that has been implemented in a tool called pathrate.**

Keywords— **Active network measurements, bandwidth monitoring, bottleneck bandwidth, available bandwidth.**

I. Introduction

The Internet is a commercial infrastructure in which users pay for their access to an Internet Service Provider (ISP), and from there to the global Internet. It is often the case that the performance level (and tariff) of these network connections is based on their bandwidth, since more bandwidth normally means higher throughput and better quality-of-service to an application. In such an environment, *bandwidth monitoring* becomes a crucial operation. Users need to check whether they get the access bandwidth that they have paid for, and whether the network 'clouds' that they use are sufficiently provisioned. ISPs also need bandwidth monitoring tools in order to plan their capacity upgrades, and to detect congested or underutilized links [4].

Network operators are increasingly using tools such as MRTG [5] to monitor the utilization of their links with information obtained from the router management software. These techniques are based on statistics maintained by the routers, and they are normally very accurate. Their drawback, however, is that they can be performed only with access to the router, and such an access is usually limited to the network manager. Instead, in this paper we focus on an *end-to-end bandwidth monitoring* approach that requires the cooperation of only the path end-points. Even though end-to-end approaches are usually not as accurate as router-based methodologies, they are often the only feasible approach for monitoring a path that crosses several networks.

We define a network path as the sequence of links that forward packets from the path sender (*source*) to the receiver (*sink*)[1]. Two bandwidth metrics that are commonly associated with a path are the *capacity* C and the *available bandwidth* A. *The capacity is the maximum IP-layer throughput that the path can provide to a flow, when there is no competing traffic load (*cross traffic*). The available bandwidth, on the other hand, is the maximum IP-layer throughput that the path can provide to a flow, given the path's current cross traffic load.* The link with the minimum transmission rate determines the capacity of the path, while the link with the minimum unused capacity limits the available bandwidth. To avoid the term *bottleneck link*, that has been widely used for both metrics, we refer to the capacity limiting link as the *narrow link*, and to the available bandwidth limiting link as the *tight link*.

Specifically, if H is the number of hops in a path, C_i is the transmission rate or *capacity* of link i, and C_0 is the transmission rate of the source, then the path's capacity is

$$C = \min_{i=0 \ldots H} C_i \quad (1)$$

Additionally, if u_i is the *utilization* of link i (with $0 \leq u_i \leq 1$ and u_0=0), the unused capacity in link i is $C_i(1-u_i)$, and so the available bandwidth of the path is

$$A = \min_{i=0 \ldots H} \left[C_i(1-u_i) \right] \quad (2)$$

Note that the available bandwidth definition requires stationary traffic and sufficiently large timescales so that the utilization terms u_i to be practically constant. The capacity and available bandwidth metrics are further discussed in the Appendix.

The *packet pair technique* is a well-known procedure to measure the capacity of a path. When a packet is transmitted in a link, it encounters a *transmission or serialization delay* due to the physical bandwidth limitations of the link and the hardware constraints of the transmitting equipment. In a link of capacity C_i and for a packet of size L, the transmission delay is $\tau_i = L/C_i$. A packet pair experiment consists of two packets sent back-to-back, i.e., with a spacing that is as short as possible, from the source to the sink. Without any cross traffic in the path, the packet pair will reach the receiver *dispersed* (spaced) by the transmission delay in the narrow link $\tau_n \equiv L/C$. So, the receiver can compute the capacity C from the measured dispersion Δ, as $C = L/\Delta$. Figure 1 illustrates the packet pair technique in the case of a three-link path, using the fluid analogy introduced in [6]. Even though simple in principle, this technique can produce widely varied estimates and erroneous results. The main reason is that the cross traffic in the path distorts the packet pair dispersion, increasing or decreasing the capacity estimates.

The main objective in this paper is to develop a capacity estimation methodology, based on end-to-end measurements, that is *robust to cross traffic effects*. We show that a straightforward application of the packet pair technique cannot, in general, produce accurate results when the cross traffic effects are ignored.

This work was supported in part by the USENIX association and by the National Science Foundation under Grant No. NCR-9711092.

[1] We assume that the path is fixed and unique, i.e., no routing changes or multipath forwarding occur during bandwidth monitoring.

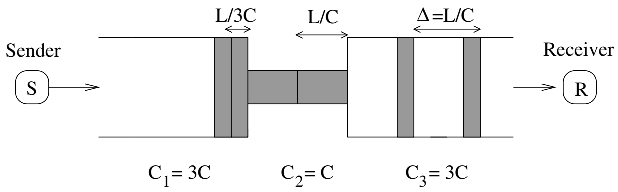


Fig. 1. Graphical illustration of the packet pair technique. The width of each link corresponds to its capacity.

The reason is that the distribution of bandwidth measurements is multimodal, and some local modes, related to the cross traffic, are often stronger than the capacity mode. The effect of the probing packet size is also investigated, showing that the conventional wisdom of using maximum sized packet pairs is not optimal in heavily loaded paths.

We then study the dispersion of long packet trains. Increasing the length of the packet train reduces the measurement variance, but the estimates converge to a value, referred to as *Asymptotic Dispersion Rate* (ADR), that is lower than the capacity. This study shows that, contrary to previous work [1], the ADR is *not* the available bandwidth in the path. For single hop paths though, we derive a formula for computing the available bandwidth from the measured ADR.

Finally, we describe a capacity estimation methodology that has been implemented in a tool called *pathrate*. This methodology uses many packet pairs to uncover the multimodal bandwidth distribution. The challenge is to identify the local modes, and to then select the mode that corresponds to the path capacity. This latter part is based on the dispersion of gradually longer packet trains. The methodology is accurate when the capacity is between 10-40 Mbps and the specified estimate resolution is 1 Mbps. For higher capacity paths, a larger estimate resolution is required.

The rest of the paper is structured as follows. Section II summarizes the previous work on bandwidth monitoring. Section III investigates the distribution of bandwidth estimates using packet pairs, while Section IV investigates the distribution of bandwidth estimates using packet trains. An analytical model for the dispersion of long packet trains is given in Section V. Section VI focuses on the size of packet pairs. Based on the insight of the previous sections, Section VII presents a capacity estimation methodology and the *pathrate* implementation. Some measurements using *pathrate* are given in Section VIII. We conclude and highlight some open problems in Section IX.

II. PREVIOUS WORK

The concept of packet dispersion, as a burst of packets crosses the narrow link of a path, was originally described in [6]. Jacobson did not consider cross traffic effects, and so the distinction between capacity and available bandwidth was not made. Keshav also studied the same idea in the context of congestion control [7], but he recognized that the dispersion of packet pairs is not related to the available bandwidth when the router queues are First-Come-First-Served (FCFS). He showed that if all routers use a fair queueing discipline, then the cross traffic is isolated and the packet pair technique can estimate the available bandwidth in the path. Bolot used packet dispersion measurements to estimate the capacity of a transatlantic link and to characterize the interarrivals of cross traffic [8].

In the past, the packet pair technique was simpler to apply. The main reason is that the possible capacity values used to be determined by a few well-known links, such as dial-up modems, ISDN links, T1's, T3's, and Ethernets. Today, mainly through the use of ATM virtual circuits/paths, the bandwidth given to a path can be any value up to the physical capacity of the underlying links. For instance, ISPs often partition an OC-3 link in several fractional virtual links, leased in a granularity of a few Mbps or so [9].

The early works on packet pair dispersion were followed by sophisticated variations, focusing on robust statistical filtering techniques. Carter and Crovella created *bprobe*, in which several packet pair measurements, originating from packets of different sizes, are processed using union and intersection filtering to produce the final capacity estimate [1]. Lai and Baker used a kernel density estimator as their statistical filtering tool [3]. In [1] and [3], the underlying assumption is that the capacity of a path is related to the most common range of bandwidth measurements, i.e., the mode of the underlying distribution. Paxson was the first to observe that the distribution of bandwidth measurements is multimodal, and he elaborated on the identification and final selection of a capacity estimate from these modes [10]. He also used packet trains of different lengths to detect multichannel links. The complete methodology is called 'Packet Bunch Modes' (PBM) [2], but as Paxson notes in his dissertation [10] (p.267): *"It is unfortunate that PBM has a large heuristic component, as it is more difficult to understand. (..) We hope that the basic ideas underlying PBM –searching for multiple modes and interpreting the ways they overlap in terms of bottleneck changes and multi-channel paths– might be revisited in the future, in an attempt to put them on a more systematic basis"*. The techniques discussed in this paper also rely on some heuristics, but contrary to Paxson's work, we explain the observed multimodalities based on cross traffic effects.

Dispersion techniques using packet trains instead of packet pairs have also been proposed for the estimation of the available bandwidth in a path. Carter and Crovella developed a tool called *cprobe* which estimates the available bandwidth from the dispersion of trains of eight packets [1]. Other researchers have proposed that the *ssthresh* variable in TCP's slow-start phase, which should ideally be set to the product of the connection's RTT with the available bandwidth, can be determined from the dispersion of the first three or four ACKs [11], [12]. The underlying assumption in [1], [11], [12] is that the dispersion of long packet trains is inversely proportional to the available bandwidth. However, as we show in this paper, this is not true.

Finally, several tools that measure the capacity of *every link in a path* were recently developed: Jacobson's *pathchar* [13], Downey's *clink* [14], Mah's *pchar* [15]; for a study of these tools see [16]. The underlying idea here is not based on the dispersion of packet pairs or trains, but on the variation of the one-way delay as the packet size increases. Unfortunately, because these tools require the generation of ICMP replies from the routers, which is a process that normally follows different processing paths in a router, the resulting measurements are often quite inaccurate. For example, a 100 Mbps Fast Ethernet link in our LAN is always measured in the range of 30-40 Mbps; similar erroneous measurements are reported in [17]. Recently, Lai and Baker proposed a technique called *packet tailgating* which avoids the need for ICMP replies from the path routers [17]. However, the reported capacity measurements are still often inaccurate. A possible explanation is that the errors in the link capacity estimates accumulate as the measurements proceed along the path.

III. PACKET PAIR DISPERSION

Consider an H-hop path defined by the sequence of capacities $\mathcal{P} = \{C_0, C_1, \ldots, C_H\}$. Two packets of size L are sent back-to-back from the source to the sink; these packets are the *packet pair* or *probing packets*. The *dispersion* of the packet pair is the interval from the instant the last bit of the first packet is received at a certain path point to the instant the last bit of the second packet is received at that point[2]. The dispersion is $\Delta_0 = \tau_0 = L/C_0$ after the source, and let it be Δ_i after link i. When the packet pair reaches the sink, the dispersion is Δ_H and the receiver computes a bandwidth estimate $b = L/\Delta_H$. Since Δ_H varies in general, if we repeat the experiment many times the b values will form a certain distribution $\mathcal{B}$. Our goal, then, is to infer a final path capacity estimate $\hat{C}$ from the distribution $\mathcal{B}$.

First, suppose that there is *no cross traffic in the path*. It is easy to see that the dispersion Δ_i cannot be lower than the dispersion at the previous hop Δ_{i-1} and the transmission delay $\tau_i = L/C_i$ at hop i, i.e., $\Delta_i = \max\{\Delta_{i-1}, \tau_i\}$. Applying this model recursively from the sink back to the source, we find that the dispersion at the receiver is

$$\Delta_H = \max_{i=0\ldots H} \tau_i = \frac{L}{\min_{i=0\ldots H}\{C_i\}} = \frac{L}{C_n} = \tau_n \qquad (3)$$

where C_n and τ_n are the capacity and the transmission delay of the narrow link, respectively. Consequently, when there is no cross traffic, all the bandwidth estimates are equal to the capacity ($b = C_n = C$).

When there is cross traffic in the path, the probing packets can experience additional queueing delays due to cross traffic. Let d_i^1 be the queueing delay of the first probing packet at hop i, and d_i^2 be the queueing delay of the second probing packet at hop i *after the first packet has been transmitted at that link* (see Figure 2). The dispersion after hop i is

$$\Delta_i = \begin{cases} \tau_i + d_i^2 & \text{if } \tau_i + d_i^1 \geq \Delta_{i-1} \\ \Delta_{i-1} + (d_i^2 - d_i^1) & \text{otherwise} \end{cases} \qquad (4)$$

Note that when $\tau_i + d_i^1 < \Delta_{i-1}$ and $d_i^2 < d_i^1$, the dispersion decreases from hop $i-1$ to hop i ($\Delta_i < \Delta_{i-1}$). This effect can cause a dispersion at the receiver that is lower than the dispersion at the narrow link, i.e., $\Delta_H < \tau_n = L/C$, if there are additional hops after the narrow link; we refer to such links as *post-narrow links*[3]. This observation means that *the capacity of the path cannot be estimated simply from the minimum measured dispersion*, as that value could have resulted from a post-narrow link.

[2] We refer to IP packet boundaries.

[3] If there are more than one links with capacity C, the narrow link is the last of them in the path.

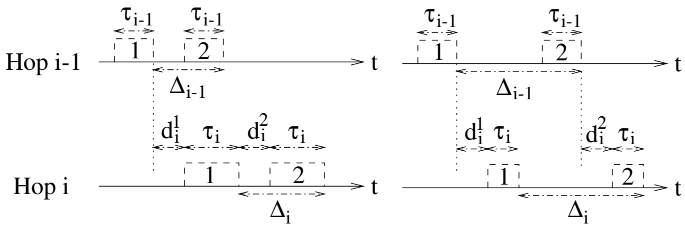
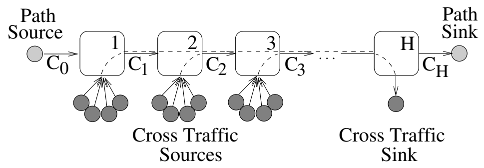
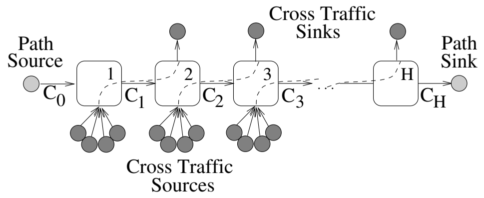


Fig. 2. The two cases of Equation 4.



(a) Path persistent cross traffic



(b) One-hop persistent cross traffic

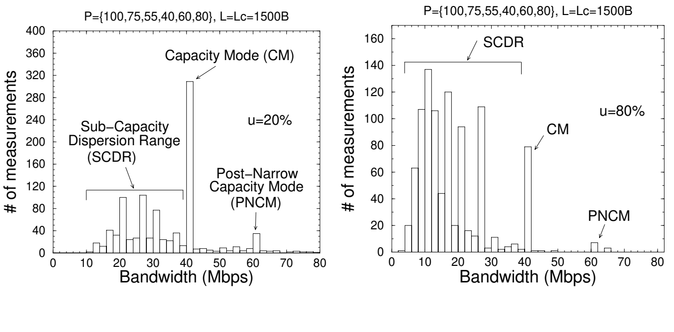
Fig. 3. The two extreme cases of cross traffic routing.

In order to examine the properties of the $\mathcal{B}$ distribution in a controllable and repeatable manner, we used the Network Simulator [18]. Simulations allow us to investigate the cross traffic effects in packet pair dispersion, avoiding issues such as route changes, multichannel links, timestamping accuracy and resolution, that can distort the measurements. We have also verified the reported results with Internet measurements[4].

The simulated model follows the description given earlier, i.e., the source sends packet pairs and the sink computes bandwidth estimates b from the measured dispersions Δ_H. The cross traffic (CT) is generated from sixteen Pareto sources at each hop with α=1.9, i.e., the interarrivals have infinite variance. The aggregation of many Pareto sources with $\alpha < 2$ has been shown to produce Long Range Dependent (LRD) traffic [19]. The CT packet size is L_c, which is either constant or follows a random distribution (described later). The packet scheduling discipline in the simulation experiments is FCFS. An important issue is the routing of the CT packets relative to the packet pairs. The two extreme cases are shown in Figure 3; in Figure 3-a the CT packets follow the same path as the packet pairs (*path persistent CT*), while in Figure 3-b the CT packets always exit one hop after they enter the path (*one-hop persistent CT*). The effect of CT routing will be discussed in § V; for now, we simulate the one-hop persistent CT case. In the following experiments, the bandwidth distribution $\mathcal{B}$ is formed from 1000 packet pair experiments.

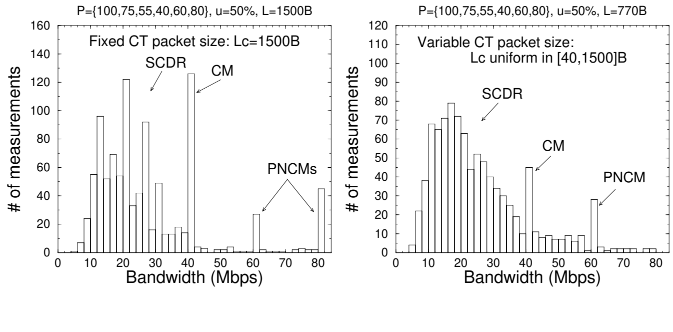
Figure 4 shows the histogram of $\mathcal{B}$, with a bin width of 2

[4] The locations of the measurement hosts are given in § VIII.



(a) Light load conditions (b) Heavy load conditions

Fig. 4. The $\mathcal{B}$ distribution in two different path loads.



(a) Fixed CT packet size (b) Variable CT packet size

Fig. 5. Fixed versus variable CT packet size L_c.

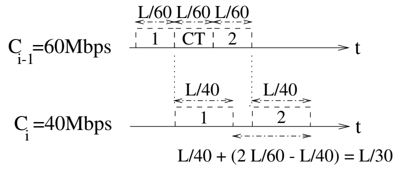


Fig. 6. Explanation of the 30 Mbps local mode in Figure 5-a.

Mbps, for a path $\mathcal{P}$ $=\{$100,75,55,40,60,80$\}$ (all capacities in Mbps). Note that the path capacity is C=40Mbps, while the post-narrow links have capacities of 60 and 80 Mbps, respectively.

In Figure 4-a, each link is 20% utilized, whereas in Figure 4-b, all links are 80% utilized. When the path is lightly loaded (u=20%) the capacity value of 40 Mbps is prevalent in $\mathcal{B}$, forming the *Capacity Mode (CM)*, which in this case is the global mode of the distribution. Bandwidth estimates that are lower than the CM are caused by CT packets that interfere with the packet pair, and they define the *Sub-Capacity Dispersion Range (SCDR)*. For instance, the SCDR in Figure 4-a is between 10 and 40 Mbps; the cause of the local modes in the SCDR is discussed in the next paragraph. Bandwidth estimates that are higher than the CM are caused in the post-narrow links when the first probing packet is delayed more than the second; these estimates are referred to as *Post-Narrow Capacity Modes (PNCMs)*. Note a PNCM at 60 Mbps, which is the capacity of the link just after the narrow link; this local mode is created when the first probing packet is delayed long enough for the packet pair to be serviced back-to-back in that link.

In heavy load conditions (u=80%), the probability of CT packets interfering with the probing packets is large, and the CM is not the global mode of $\mathcal{B}$. Instead, the global mode is in the SCDR, which now dominates the bandwidth measurements. A key point here is that *the path capacity cannot be always correctly estimated by statistical techniques that extract the most common bandwidth value or range*. Instead, we must examine the resulting bandwidth distribution in queueing terms, analyze what causes each of the local modes, and what differentiates the CM from the rest of the local modes.

Figure 5 shows $\mathcal{B}$ for the same path when the CT packet size L_c is fixed (1500 bytes) and when it varies uniformly in the range [40, 1500] bytes (u=50%). In the first case, the probing packet size L is also 1500 bytes, while in the second case it is 770 bytes, i.e., the average of the [40, 1500] range[5]. When all packets have the same size (L_c=L=1500B), it is simpler to explain the local modes in the SCDR. For instance, consider the path $\mathcal{P}$ $=\{100, 60, 40\}$, and assume that all packets have the same size. A local mode at 30 Mbps can be caused by a packet interfering with the packet pair at the 60 Mbps link, since in that case the dispersion after the narrow link is $\Delta_n = \frac{L}{40} + (2\frac{L}{60} - \frac{L}{40}) = \frac{L}{30}$ (see Figure 6). Similarly, a mode at 20 Mbps is caused by a packet interfering with the packet pair at the 40 Mbps link or by two packets interfering at the 60 Mbps link, and so on.

When the CT packet size varies uniformly in the range [40,1500]B though (Figure 5-b), the resulting dispersion is less predictable, since a single packet interfering with the packet pair can produce a range of dispersion values, depending on its size. However, *the CM and one or more of the PNCMs are still distinct in the distribution*, as they are caused by the probing packets being serviced back-to-back from the narrow or from post-narrow links, respectively.

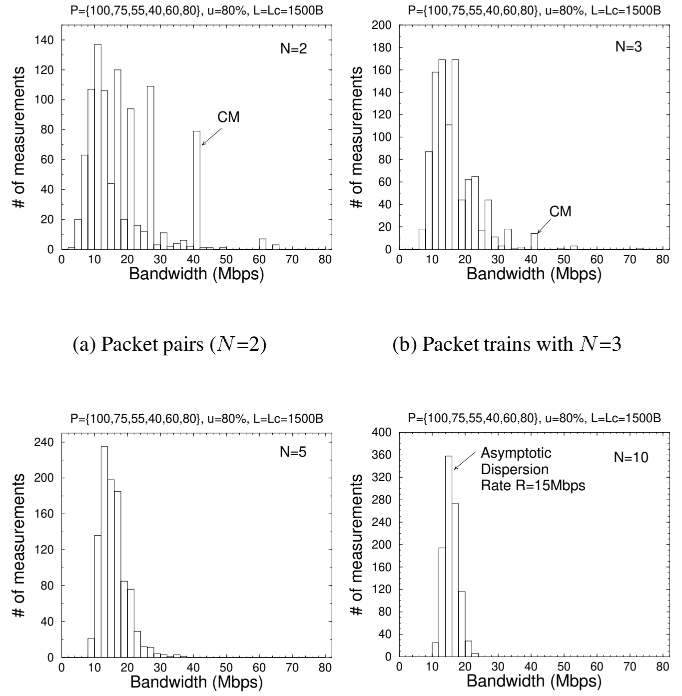
Several measurement studies have shown that the packet size distribution in the Internet is centered around three or four values [20], [21]. Specifically, about 50% of the packets are 40 bytes, 20% are 552 or 576 bytes, and 15% are 1500 bytes. These common packet sizes would cause a packet pair bandwidth distribution that is more similar to the 'discrete dispersion' effects of Figure 5-a, rather than the 'continuous dispersion' effects of Figure 5-b.

IV. Packet train dispersion

Extending the packet pair technique, the source can send $N > 2$ back-to-back packets of size L to the sink; we refer to these packets as a *packet train of length* N. The sink measures the total dispersion $\Delta(N)$ of the packet train, from the first to the last packet, and computes a bandwidth estimate as $b(N) = \frac{(N-1)L}{\Delta(N)}$. Many such experiments form the bandwidth distribution $\mathcal{B}(N)$.

If there is no cross traffic in the path, the bandwidth estimates will be equal to the capacity C, as in the packet pair case. Measuring the capacity of a path using packet trains is required when the narrow link is multichanneled [2]. In a k-channel link of total capacity C, the individual channels forward packets in parallel at a rate of C/k and the link capacity can be measured from

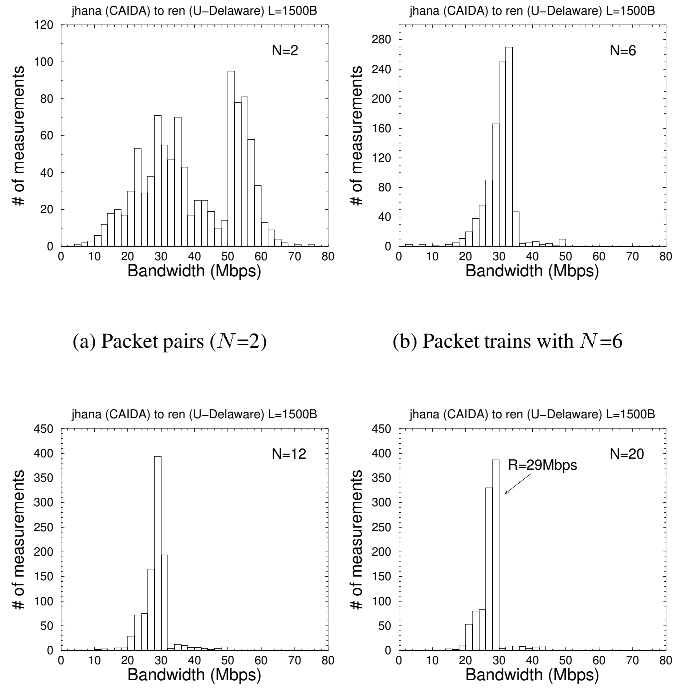
[5] More about the selection of L in § VI.



(a) Packet pairs (N=2) (b) Packet trains with N=3

(c) Packet trains with N=5 (d) Packet trains with N=10

Fig. 7. The effect of the packet train length (simulations).



(a) Packet pairs (N=2) (b) Packet trains with N=6

(c) Packet trains with N=12 (d) Packet trains with N=20

Fig. 8. The effect of the packet train length (measurements).

the dispersion of packet trains with N=k+1. Packet trains are also required to measure the *sustainable rate* of a traffic shaper[6].

It may appear at first that using packet trains, instead of packet pairs, makes the capacity estimation more robust to random noise caused by cross traffic. One can argue that this is true because packet trains lead to larger dispersion values, which are more robust to measurement noise. However, this is not the case due to the following reason. Although the dispersion $\Delta(N)$ becomes larger as N increases, so does the 'noise' in the measured values of $\Delta(N)$, since it becomes more likely that CT packets will interfere in the packet train. This issue was also briefly mentioned in [10] (p.259), noting that packet trains should be less prone to noise, since individual packet variations are smoothed over a single large interval rather than N-1 small intervals, *but* with a larger N the greater the likelihood that a packet train will be dispersed by cross traffic, leading to bandwidth underestimation.

In this section, we present simulation and experimental results illustrating the effect of N in the bandwidth distribution $\mathcal{B}(N)$, and make some general observations about this relation. Figure 7 shows the histograms of $\mathcal{B}(N)$, for four increasing values of N, from simulations of the path $\mathcal{P}$ ={100,75,55,40,60,80} with u=80% in all links. Figure 8 shows the histograms of $\mathcal{B}(N)$, for four increasing values of N, from Internet measurements at the path from *jhana* (in San Diego CA) to *ren* (in Newark DE) during June 2000.

A first observation is that, *as N increases, the CM and PCNMs become weaker, until they disappear, and the SCDR prevails in the bandwidth distribution $\mathcal{B}(N)$*. The reason is that, as N increases, almost all packet trains encounter additional dispersion due to CT packets. This also means that *the best value of N for generating a strong capacity mode is N=2, i.e., to use packet pairs; anything longer than packet pairs is more likely to get additional dispersion due to cross traffic*.

A second observation is that, *as N increases, $\mathcal{B}(N)$ becomes unimodal*. This implies that, when N is large, the dispersion of packet trains by CT packets is not determined by distinct interference cases, forming local modes, but it is determined by the aggregate amount of CT interfering with the packet train.

A third observation is that *the range of the distribution, which is related to the measurement variance, decreases as N increases*. This means that the variance in the amount of cross traffic interfering with the packet train decreases, as the length of the packet train increases.

A fourth observation is that, *when N is sufficiently large and $\mathcal{B}(N)$ is unimodal, the center of the (unique) mode is independent on N*. We refer to the center of this unique mode as the *Asymptotic Dispersion Rate (ADR) R*. The fact that ADR does not depend on the packet train length means that, for sufficiently large N, the dispersion of the packet train $\Delta(N)$ becomes proportional to N-1, and thus the packet train length cancels out from the bandwidth estimate $b(N) = \frac{(N-1)L}{\Delta(N)}$; this observation is explained in the next section for certain special cases.

V. Asymptotic Dispersion Rate

In this section, we present a model for the dispersion of packet trains, taking into account the cross traffic in the path. First, consider a single hop path $\mathcal{P}$= $\{C_0, C_1\}$ with $C_0 \geq C_1$, i.e., the

[6] Traffic shapers, usually in the form of a leaky bucket, limit the capacity of a (virtual) link from a peak rate to a sustainable rate after a certain burst size.

C_1 link ('link-1') is the narrow link. A packet train of length N is sent from the source to the sink with initial dispersion $\Delta_0 = L(N-1)/C_0$. Let r_1 be the average incoming rate of cross traffic in link-1. The average amount of cross traffic that arrives in link-1 during Δ_0 is $\bar{X}_1 = \Delta_0 r_1$. Assuming that the link-1 queue is serviced in a FCFS basis, the $\bar{X}_1$ cross traffic interferes with the packet train packets, and so the average dispersion at the exit of the narrow link is

$$\bar{\Delta}_1 = \frac{(N-1)L + \bar{X}_1}{C_1} = \frac{(N-1)L}{C_1}(1 + u_1\frac{C_1}{C_0}) \quad (5)$$

where $u_1 = r_1/C_1$ is the load (utilization) of the narrow link due to cross traffic.

Consequently, the average bandwidth estimate at the receiver, that we refer to as the Asymptotic Dispersion Rate R, is

$$R \equiv \frac{(N-1)L}{\bar{\Delta}_1} = \frac{C_1}{1 + u_1\frac{C_1}{C_0}} < C_1 \quad (6)$$

which is lower than the path capacity. Note that *the ADR is independent of* N, as noted in § IV, since the amount of interfering cross traffic $\bar{X}_1$, and thus the overall dispersion $\bar{\Delta}_1$, is proportional to N-1. As shown in Figures 7-d and 8-d, even with the bursty Pareto cross traffic or with the actual Internet traffic, a value of N around 10-20 is normally sufficient to produce a narrow estimate of R.

Some comments on Equation 6 follow. First, if the capacities C_0 and C_1 are known, we can measure R from the dispersion of long packet trains, compute the cross traffic utilization u_1 from Equation 6, and then compute the available bandwidth as $A = C_1(1-u_1)$. So, the available bandwidth of single hop paths can be estimated, using the dispersion of packet trains that are sufficiently long to produce a narrow estimate of R. This also implies that the available bandwidth is not inversely proportional to the dispersion of long packet trains, as was assumed in [1], even for single hop paths. For example, in the path of Figure 7-d we have that R=15 Mbps, while A=40(1-0.8)=8 Mbps. Second, for capacity estimation purposes, it helps to 'inject' the probing packets in the path from a higher bandwidth interface (higher C_0), since the cross traffic term $u_1\, C_1/C_0$ is then smaller. Third, the term $u_1\, C_1/C_0$ is equal to $\bar{X}_1/[(N-1)L]$, and so, it is equal to the average number of CT bytes interfering with two successive probing packets.

These results can be generalized to an H-hop path with $C_0 \geq C_1 \geq \ldots \geq C_H$, for the case of path persistent cross traffic (§III). Let r_i be the average rate of cross-traffic that enters the path in link i[7]. The average dispersion at the exit of link i, then, is $\bar{\Delta}_i = \bar{\Delta}_{i-1}(C_{i-1} + r_i)/C_i$, and the ADR becomes

$$R = \frac{(N-1)L}{\bar{\Delta}_H} = \frac{C_H}{\prod_{i=1}^{H}(1 + \frac{r_i}{C_{i-1}})} \quad (7)$$

For instance, for the path $\mathcal{P}= \{C_0, C_1, C_2\}$ with $C_0 \geq C_1 \geq C_2$:

$$R = \frac{C_2}{1 + \frac{r_1}{C_0} + \frac{r_2}{C_1} + \frac{r_1 r_2}{C_0 C_1}} \quad (8)$$

[7] Since the cross traffic is path persistent (see Figure 3-a), the total cross traffic rate in link i is $\sum_{k=1}^{i} r_k$.

When the capacities do not decrease along the path, the analysis is more complicated. In the single-hop case $\mathcal{P}= \{C_0, C_1\}$ with $C_0 < C_1$, there would be an idle spacing of duration $L/C_0 - L/C_1$ at the exit of link-1 between any two probing packets, if there was no cross traffic. The cross traffic can fill in the idle space in the packet train, or cause additional dispersion without filling in all the idle space. A lower bound on the dispersion Δ_1 can be derived if we assume that the cross traffic increases the packet train dispersion beyond Δ_0 only after it fills in all the idle spacing. When this is the case, the dispersion at the receiver is

$$\bar{\Delta}_1 = \Delta_0 + \max\left\{\frac{\bar{X}_1}{C_1} - (N-1)L(\frac{1}{C_0} - \frac{1}{C_1}), 0\right\} \quad (9)$$

If the cross traffic load is sufficiently low ($r_1 < C_1 - C_0$), the dispersion is not increased at link-1 (i.e., $\bar{\Delta}_1$=Δ_0), and so $R = C_0$. Otherwise, the final dispersion becomes $\bar{\Delta}_1 = \frac{(N-1)L}{C_0}(u_1 + \frac{C_0}{C_1})$, which gives the same ADR value as Equation 6.

These results can be extended for the case of H hops, when the cross traffic is path persistent. Specifically, a lower bound on the dispersion Δ_H can be derived if we assume that the cross traffic increases the packet train dispersion only after it fills in all the idle spacing between probing packets. Then,

$$\bar{\Delta}_i = \begin{cases} \bar{\Delta}_{i-1}\frac{C_{i-1}+r_i}{C_i} & \text{if } C_{i-1} \geq C_i \\ & \text{or } r_i \geq C_i - C_{i-1} > 0 \\ \bar{\Delta}_{i-1} & r_i < C_i - C_{i-1} \end{cases} \quad (10)$$

Given the capacities and cross traffic rates in each hop, and since $\Delta_0 = L(N-1)/C_0$, we can solve recursively for $\bar{\Delta}_H$, and thus for R.

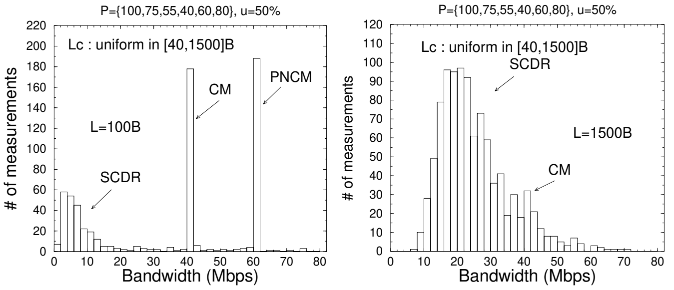
When the cross traffic is not path persistent, i.e., CT packets exit the path before the last hop, the dispersion of packet trains is hard to analyze for the same reason: CT packets can interfere in the packet train increasing its dispersion, and then exit the path leaving idle space, or 'bubbles', between probing packets. These bubbles can be filled in by CT packets in subsequent hops, or they can persist until the packet train reaches the sink. For the case of one-hop persistent cross traffic (see Figure 3-b), an upper and a lower bound can be derived for R. Note that since the cross traffic is assumed to be one-hop persistent in this case, the utilization of link i is $u_i = r_i/C_i$. For an H-hop path in which all links have the same capacity C, it can be shown that the ADR is

$$\frac{C}{\prod_{i=1}^{H}(1 + u_i)} \leq R \leq \frac{C}{1 + \max_{i=1\ldots H} u_i} \quad (11)$$

The lower bound corresponds to the case that bubbles are never filled in, while the upper bound corresponds to the case that the bubbles created at the link with the maximum utilization are the only ones that reach the receiver, and that the rest of the path links just fill in (partially) those bubbles.

VI. THE SIZE OF PROBING PACKETS

In this section, we focus on the effect of the packet size L in packet pair probing. The 'conventional wisdom', as reflected



(a) L=100 bytes (b) L=1500 bytes

Fig. 9. Small versus large packet size in packet pair probing.

for instance in [1] or [10], is that the optimal L is the maximum non-fragmented packet size, i.e., the *path Maximum Transmission Unit (MTU)* size. The reason is that a higher L leads to larger dispersion, which in turn is easier to measure, more robust to queueing delay noise, and less sensitive to the timestamping resolution at the receiver.

This previous reasoning, however, does not take into account the effects of cross traffic. A larger packet size L leads to a wider time interval in which a CT packet can interfere with the packet pair. Suppose that a packet pair arrives at a link i with capacity C_i. If a CT packet arrives at link i in the time interval between the arrival of the first and the second probing packets, which is of length L/C_i, it will interfere with the probing packets, increasing the dispersion Δ_i above τ_i. The larger the L, the higher the likelihood of an interfering CT arrival, and thus the SCDR becomes more prevalent in the bandwidth distribution $\mathcal{B}$. This effect is shown in Figure 9, where $\mathcal{B}$ is shown for a small packet (L=100B), versus a large, Ethernet frame sized, packet (L=1500B). The narrow link dispersion τ_n is 15 times smaller in the L=100B case, causing a much weaker SCDR than the L=1500B case.

A minimum sized packet, however, is not optimal either. As L decreases, the dispersion decreases proportionally, and thus, it becomes more susceptible to distortion at the post-narrow links. Suppose that L=100B, $\mathcal{P}$=$\{40,80\}$ and that a packet pair leaves the narrow link back-to-back, i.e., with Δ_0=20μs. It only takes one CT packet, larger than 100 bytes, at the 80 Mbps link to delay the first probing packet so much that the packet pair dispersion is controlled by that link, i.e., Δ_1=10μs. In other words, when L is small, the formation of PNCMs becomes more likely and the CM becomes weaker. This can be seen in Figure 9-a; note the strong PNCM at 60 Mbps, which is actually stronger than the CM at 40 Mbps. On the other hand, there are no significant PNCMs when L=1500B, as shown in Figure 9-b.

Given the previous trade-off in the selection of the packet size, a value of L somewhere in the middle of the L_c range is preferred. For instance, compare Figure 9 with the bandwidth distribution in Figure 5-b, where L is set to the average of the CT packet size range (L=770B): the CM is strong in Figure 5-b compared to both the SCDR and PNCM parts of the bandwidth distribution. The empirical conclusion from our Internet experiments is that a packet size around 800 bytes leads to the stronger CM in heavily loaded paths. For lightly loaded paths, the selection of the packet size is not so important.

Finally, we note a practical issue that is related to the *minimum dispersion that the receiver can measure*. A receiving host can only measure the dispersion of a packet pair when it is higher than Δ_m. This lower bound Δ_m is determined by the latency to receive a packet in the OS, to move the packet from kernel to user space through a *recvfrom* system call, to timestamp the arrival, and to perform any other operations of the receiving program before waiting for the second probing packet. For *pathrate*, we measured Δ_m in several different platforms, including Sun Ultra-10 and Pentium-II workstations running Solaris 2.6 or Free-BSD 3.2, and the minimum dispersion Δ_m is in the order of 30 to 40 μs. Given Δ_m for a specific receiver, the maximum possible capacity that can be measured for a packet size L is $C = L/\Delta_m$. For example, with Δ_m=40μs and L=800B, the maximum capacity that can be measured is 160 Mbps. On the other hand, when a rough estimate $\bar{C}$ of the capacity is known, the packet size should be at least $L > \bar{C}\Delta_m$.

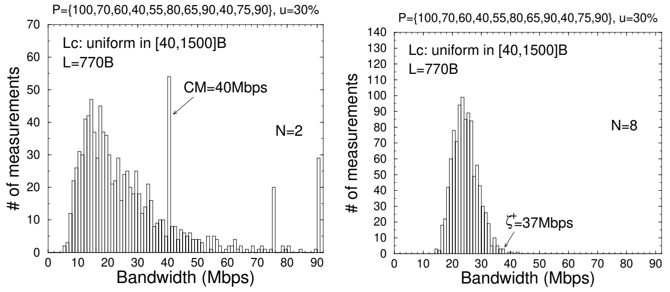
VII. A CAPACITY ESTIMATION METHODOLOGY

In this section, we present a capacity estimation methodology based on the insight developed so far in the paper. This methodology has been implemented in a tool called *pathrate*. The *pathrate* methodology requires the cooperation of both the source and the sink, i.e., it is a *two end-point methodology*. More flexible approaches require access only at the source of the path, 'forcing' the sink to reply to each received packet using ICMP, UDP-echo, or TCP-FIN packets. The problem in those approaches is that the reverse path from the sink to the source, through which the replies are forwarded, affects the bandwidth measurements, making it hard to decouple the characteristics of the two paths. We prefer the two end-point methodology, even though it is less flexible, because it is more accurate.

Phase I: Packet pair probing. As shown in § IV, one is more likely to observe the capacity mode using packet pairs than using packet trains. Consequently, in this phase we use a large number of packet pair experiments to 'uncover' all the local modes of the bandwidth distribution $\mathcal{B}$, expecting that one of them is the CM. Also, as shown in § VI, there is a trade-off in the selection of the probing packet size L: smaller packets lead to stronger PNCMs, while larger packets lead to a more prevalent SCDR. A probing packet size of L=800 bytes usually leads to the strongest CM in the resulting bandwidth distribution. In *pathrate*, Phase I consists of K_1=2000 packet pair experiments using a packet size of L=800 bytes.

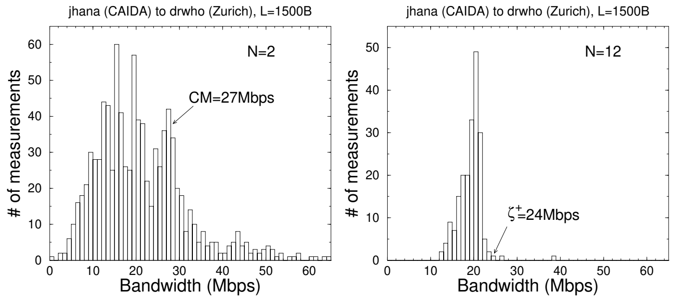
From the resulting distribution of bandwidth measurements $\mathcal{B}$, we obtain all the local modes. The numerical procedure for the identification of the local modes is not described here due to space constraints. It is similar to the algorithm described in [10], but the user has to specify the *histogram bin width* ω, which is also the *resolution of the final capacity estimate*. If, for example, the resolution is ω=2 Mbps, *pathrate* will produce a final estimate that is a 2 Mbps interval. As will be shown later, the resolution is a critical parameter for the accuracy of the final result.

The sequence of local modes, *in increasing order*, is denoted as $\mathcal{M}$= $\{m_1, m_2, \ldots m_M\}$. We expect that one of these local modes, say m_k, is the CM (i.e., $C = m_k$), with the larger modes



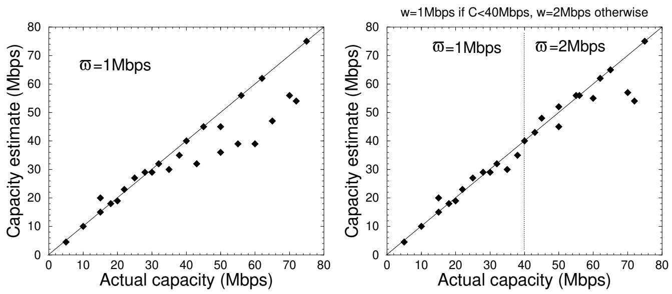
(a) N=2 (b) N=$\bar{N}$=8

Fig. 10. Illustration of capacity estimation (simulations).



(a) N=2 (b) N=$\bar{N}$=12

Fig. 11. Illustration of capacity estimation (measurements).



(a) One bin width (b) Two bin widths

Fig. 12. Evaluation of the heuristic of Equation 12.

being PCNMs, and the smaller modes being in the SCDR of $\mathcal{B}$. If the distribution $\mathcal{B}$ is unimodal, which happens in very lightly loaded paths, the measurement process terminates and the capacity estimate $\hat{C}$ is the unique mode m_1. Otherwise, Phase-II selects m_k from $\mathcal{M}$.

Phase II: Packet train probing. As noted in § IV, as N increases, the CM and the PNCMs are eliminated from the bandwidth distribution $\mathcal{B}(N)$, and the SCDR accumulates all measurements. Gradually, $\mathcal{B}(N)$ becomes unimodal, centered at the Asymptotic Dispersion Rate R, and the width of this unique mode is reduced as N increases. Let $\bar{N}$ be the *minimum* value of N for which $\mathcal{B}(N)$ is unimodal. Also, let $[\zeta^-, \zeta^+]$ be the range of the unique mode, i.e., the bandwidth interval that includes all the significant values in the 'bell' around R[8]. The heuristic rule with which the capacity estimate $\hat{C}$ is selected is that *the capacity mode is the minimum mode m_i in $\mathcal{M}$ which is higher than* ζ^+, i.e.,

$$\hat{C} = m_k = \min\{m_i \in \mathcal{M} : m_i > \zeta^+\} \tag{12}$$

The heuristic is based on the following reasoning. First, when N is sufficiently large for $\mathcal{B}(N)$ to be unimodal, almost all packet trains have encountered dispersion due to cross traffic, and so $\zeta^+ < C$. Second, because $\bar{N}$ is the *minimum* packet train length that generates a unimodal $\mathcal{B}(N)$, the range of the unique mode is still sufficiently wide to cover all the local modes in the SCDR of $\mathcal{B}$ between R and C. This heuristic resulted from long experimentation, and is evaluated later in this section.

In *pathrate*, Phase II consists of a K_2=400 packet train experiments with L=1500B for each length N. If the resulting distribution $\mathcal{B}(N)$ is not unimodal, N is increased by two, and the process repeats. Note that K_2 is significantly lower than K_1, because in Phase II we only check whether the distribution is unimodal, instead of estimating the local modes. When the length $N = \bar{N}$ is reached, the upper threshold ζ^+ is measured, and the capacity estimate m_k is determined from Equation 12.

To illustrate the use of Equation 12, Figures 10 and 11 show the packet pair distribution $\mathcal{B}$ and the unimodal distribution $\mathcal{B}(\bar{N})$ for a simulation and a real network experiment, respectively. The distributions in Figure 10 result from simulating the path $\mathcal{P}= \{100, 70, 60, 40, 55, 80, 65, 90, 40, 75, 90\}$ with one-hop persistent cross traffic. The sequence of modes in $\mathcal{B}$, with ω=1 Mbps, is $\mathcal{M}$={9,14,17,23,26,29,33,40,44,56,75,90}. The minimum N that results in a unimodal distribution is $\bar{N}$=8, and the upper threshold of the mode is ζ^+=37 Mbps. Consequently, from Equation 12, the estimated capacity is $\hat{C} \approx 40$ Mbps, which is the correct value.

[8]The exact algorithm for the estimation of $[\zeta^-, \zeta^+]$ involves heuristics to separate measurements in the R 'bell' from measurement noise.

The distributions in Figure 11 result from experiments in a network path from San Diego to Zurich (see § VIII). The sequence of modes in $\mathcal{B}$ is $\mathcal{M}$={9,11,13,15.5, 19.5, 27, 32, 43}, $\bar{N}$=12, and ζ^+=24 Mbps. So, the estimated capacity is $\hat{C} \approx 27$ Mbps, which is the correct value (see § VIII).

In order to evaluate the accuracy of the presented heuristic, we simulated the *pathrate* methodology in a number of different path and traffic configurations, comparing the actual capacity C with the capacity estimate $\hat{C}$. The simulated cases cover a range of values for H (3 to 15), C_i (5 to 125 Mbps), C (5 to 75 Mbps), u_i (0.1 to 0.9), L_c (constant or uniformly distributed in [40,1500]B), and one-hop or path persistent cross traffic. Figure 12-a shows the results when ω=1 Mbps. The methodology is quite accurate, leading to $\hat{C} \approx C$, as long as the path capacity is lower than about 40 Mbps.

For higher path capacities, $\hat{C}$ is lower than C, in some cases by almost a factor of two. It turns out that these erroneously low estimates are usually the first local mode in the SCDR of $\mathcal{B}$ that is lower than the CM, and so the assumption that the unique mode in $\mathcal{B}(\bar{N})$ includes all the SCDR modes between R and C is not always true. This mainly occurs in paths with heavy load (more than 80%) in the narrow or pre-narrow links.

Figure 12-b shows the results of the same simulations, but

when the specified bin width ω is 1 Mbps for lower capacity paths ($C < 40$ Mbps), and 2 Mbps for higher capacity paths ($C > 40$ Mbps). Note that the estimates are more accurate for the high capacity paths with the larger bin width. The few estimates that are still too low are corrected with an even larger bin width (ω=3 Mbps), at the cost of a wider resolution. The bin width has this effect because, as ω increases, the weak modes in the SCDR of $\mathcal{B}$ which are close to the CM, and which cause the underestimations, tend to merge with the capacity mode. If ω is too large, on the other hand, the CM can merge with the SCDR modes and the final estimate will be one of the PNCMs (overestimation). In other words, the resolution ω has to be chosen based on a rough estimate of the path's capacity. More work is needed for an adaptive selection of ω.

There are several features and issues about *pathrate*, that we only briefly mention here. Before Phase I, *pathrate* generates packet trains of gradually increasing length to detect multichannel links; if there is a steep bandwidth decrease when N increases from k to $k+1$, we infer that the narrow link consists of k channels. This initial set of packet trains is also used to determine the maximum packet train length that the path can transfer without causing buffer overflows at the routers or the sender/receiver OS. Note that we avoid packet trains that are too long and cause buffer overflows in order to not affect the cross traffic, which normally responds to losses using the congestion avoidance mechanisms of TCP.

pathrate uses UDP for the probing packets. Additionally, it establishes a TCP connection, referred to as the *control channel*, which acknowledges every correctly received packet pair/train, and is used for exchange of control information between the two end-points. Any packet pairs or trains that encountered losses are ignored from the measurement process. As a simple form of congestion avoidance, *pathrate* aborts the measurement process when it detects significant losses in the path. The time interval between successive packet pairs or trains is set to 500 msec; so, when *pathrate* sends packet trains with L=1500B and N=10, the average rate of probing traffic is 240 kbps.

Currently, the receiving part of *pathrate* uses user-level timestamping. This often causes bandwidth estimates that are higher than the bandwidth of the network interface at the receiving host, because two closely received packets can be queued at the kernel and then delivered to the application with a small spacing that is indicative of the kernel-user bandwidth. These estimates do not normally cause errors, since they produce very large modes in $\mathcal{M}$ which are unlikely to be selected as m_k. If the receiver's network interface bandwidth (that is C_H) is known, we know that the measurements that are higher than C_H have been caused at the receiving host end, and so we can 'clamp' them to C_H.

VIII. Capacity measurements

In this section, we present a few capacity measurements using *pathrate* in a mesh of five hosts in US and Europe. The host names and their geographical location are shown in Table I. The paths between these hosts cross several academic and commercial networks, such as the vBNS, Abilene, Dante, CalRen2, UUnet, Cable & Wireless, Switch, and the local access networks at each site. *zamboni* is connected to a 10 Mbps Ethernet interface, while the rest of the hosts are connected to Fast Ethernet interfaces (100 Mbps).

TABLE I

Measurement hosts and their locations.

Host	Location
sun	Univ.Wisconsin, Madison WI
jhana	CAIDA, San Diego CA
zamboni	CMU, Pittsburgh PA
ren	Univ.Delaware, Newark DE
drwho	ETH, Zurich-Switzerland

TABLE II

Capacity measurements with *pathrate*

	sun	*jhana*	*zamboni*	*ren*	*drwho*
sun	100	100-104	9-10	90-94	28-29
jhana	108-112	100	9-10	106-110	27-28
zamboni	9-10	9-10	10	13-14	26-27
ren	108-112	98-102	9-10	100	26-27
drwho	25-26	26-27	26-27	26-27	100

The *pathrate* capacity measurements are shown in Table II. The measurements in the row of a host refer to the capacities of the paths that *originate* from that host. For instance, the capacity estimate for the path from *sun* to *drwho* is 28-29 Mbps. The bin width selection was an 'educated guess', in the sense that ω was set to 1 Mbps when the bandwidth measurements were mostly below 50 Mbps, and to 4 Mbps when the measurements were higher. Specifically, all paths that involve *zamboni* or *drwho* were measured with ω=1 Mbps, while the rest of the paths were measured with ω=4 Mbps. The measurements were performed during weekdays and daytime at both ends of the path. The measurements that involve *drwho* were performed during June 2000; at that time *zamboni* was still connected to a Fast Ethernet. The rest of the measurements were performed during December 2000, while preparing the final version of this paper.

We verified some of these measurements, by contacting the network managers of the involved sites. Specifically, in June 2000 *drwho* was still connected to US through a transatlantic 32 Mbps ATM UUnet link operated by Switch[9]. Due to the involved AAL5 and ATM header overheads, the IP-layer capacity of the link is about 28.3 Mbps for 1500B packets, and about 27.4 Mbps for 500B packets. As shown in Table II, the *pathrate* measurements are quite close to this value, in the range 25-29 Mbps. *pathrate* accurately measures the 10 Mbps capacity of the paths that are connected to *zamboni*, with the exception of the capacity in the path from *zamboni* to *ren* which is slightly overestimated ($\hat{C}$=13-14 Mbps). Unfortunately, we were unable to verify the rest of the capacity measurements due to insufficient information about the involved networks. The paths between *sun*, *jhana*, and *ren* though, lead to results in the range 90-110 Mbps, implying that the corresponding paths may be limited by the Fast Ethernet network interfaces (100 Mbps) of the measurement hosts. This is likely to be the case, since the corresponding

[9] In fact, the link consisted of two 32 Mbps ATM virtual paths, but a certain microflow could only use one of the two VPs. Later in the summer of 2000 that link was upgraded to a POS OC-3.

institutions (University of Wisconsin, CAIDA, and University of Delaware) are connected to the their network providers through ATM or POS OC-3 links (about 140-155 Mbps).

IX. CONCLUSIONS

This paper studied the dispersion of packet pairs and packet trains, focusing on the effects of the cross traffic. As an application of this study we developed a capacity estimation methodology. The insight gained, though, can probably be also applied to congestion control mechanisms, server selection algorithms, as well as quality of service monitoring. A first task for future work is to improve the capacity estimation methodology, and specifically the heuristic specified by Equation 12, so that the underestimation errors shown in Figure 12 are avoided. This is also related to the selection of the bin width or resolution ω. We also examine the sensitivity of the results to the cross traffic load, running *pathrate* in different times of day. Finally, the ADR metric, which is related to the utilization of all links in the path, may be a useful metric for monitoring the quality of service that the path offers, and it would be interesting to examine its dynamic variations over both short and long timescales.

APPENDIX

The definition of the path capacity C in Equation 1 is straightforward. There are two points, though, that one has to be careful with. First, the use of additional headers in layer-2 technologies can result in an IP-layer capacity that is lower than the 'advertized' nominal bandwidth of a link. Second, in certain multichannel links the router performs hashing based on the destination address of the packet, or based on the 5-tuple header fields, in order to determine the specific sub-link that the packet will be forwarded to. In that case, all the probing packets will be sent to the same sub-link, and so the measurement tool will measure the capacity of only that sub-link. This is also, however, the maximum throughput that a certain IP microflow would be able to get in the path.

Regarding the available bandwidth A, defined in Equation 2, we make the following remarks. A is the maximum available throughput for a *congestion responsive* flow, i.e., a flow that does not attempt to 'steal' bandwidth from the cross traffic. Obviously, a congestion unresponsive flow can get a higher throughput than A if it attempts to saturate the path, causing losses in the TCP part of the cross traffic. Sometimes the available bandwidth is defined as the sustained throughput of a long TCP connection in the path [22]. The TCP throughput, however, depends on the version and the specific implementation of the TCP congestion avoidance mechanisms [23]. Also, the throughput of a long TCP connection ('elephant') is not the same with the aggregate throughput of a large number of short TCP connections ('mice') in the same path and load conditions. For these reasons, we believe that it is more appropriate to define the available bandwidth in terms of the load (utilization) of the path links, as in Equation 2. The available bandwidth A, then, has to be interpreted as the maximum throughput that a congestion responsive flow would get, if the flow was able to saturate the tight link in the path, but without causing any reduction in the cross traffic load.

ACKNOWLEDGMENTS

We are grateful to the following people for providing us with computer accounts at their sites: Tobias Oetiker (ETH), Andy Myers and Hui Zhang (CMU), Adarsh Sethi and Paul Amer (Univ-Delaware), Hans-Werner Braun (NLANR), David Meyer (Univ-Oregon). We are also grateful to Jambi Ganbar from the vBNS network engineering group for experimenting with *pathrate* in the vBNS, to Simon Leinen from the Switch network in Switzerland for crucial information about their transatlantic link, to Daniel Grim for information regarding the University of Delaware Internet access, and to Allen Downey, Kevin Lai, Bob Melander, and the Infocom reviewers for providing useful comments on this paper. Finally, we are grateful to Kimberly Claffy, Tracie Monk, Evi Nemeth, and all the CAIDA 'elves' for their useful comments and help in completing this work.

REFERENCES

[1] R.L. Carter and M.E.Crovella, "Measuring Bottleneck Link Speed in Packet-Switched Networks," *Performance Evaluation*, vol. 27,28, pp. 297–318, 1996.

[2] V. Paxson, "End-to-End Internet Packet Dynamics," *IEEE/ACM Transaction on Networking*, vol. 7, no. 3, pp. 277–292, June 1999.

[3] K. Lai and M.Baker, "Measuring Bandwidth," in *Proceedings IEEE INFOCOM*, Apr. 1999.

[4] R. Caceres, N.Duffield, and A.Feldmann, "Measurement and Analysis of IP Network Usage and Behavior," *IEEE Communications Magazine*, pp. 144–152, May 2000.

[5] T. Oetiker, "MRTG: Multi Router Traffic Grapher," http://ee-staff.ethz.ch/∼oetiker/webtools/mrtg/mrtg.html.

[6] V. Jacobson, "Congestion Avoidance and Control," in *Proceedings ACM SIGCOMM*, Sept. 1988, pp. 314–329.

[7] S. Keshav, "A Control-Theoretic Approach to Flow Control," in *Proceedings ACM SIGCOMM*, Sept. 1991.

[8] J. C. Bolot, "Characterizing End-to-End Packet Delay and Loss in the Internet," in *Proceedings ACM SIGCOMM*, 1993, pp. 289–298.

[9] UUNET, "UUNET Technologies," http://www.uunet.net/, Nov. 2000.

[10] V. Paxson, *Measurements and Analysis of End-to-End Internet Dynamics*, Ph.D. thesis, University of California, Berkeley, Apr. 1997.

[11] J.C. Hoe, "Improving the Start-up Behavior of a Congestion Control Scheme for TCP," in *Proceedings ACM SIGCOMM*, Sept. 1996.

[12] L. S. Brakmo and L.L.Peterson, "TCP Vegas: End to End Congestion Avoidance on a Global Internet," *IEEE Journal on Selected Areas of Communications*, vol. 13, no. 8, Oct. 1995.

[13] V. Jacobson, "pathchar: A Tool to Infer Characteristics of Internet Paths," ftp://ftp.ee.lbl.gov/pathchar/, Apr. 1997.

[14] A.B. Downey, "clink: a Tool for Estimating Internet Link Characteristics," http://rocky.wellesley.edu/downey/clink/, June 1999.

[15] B. A. Mah, "pchar: a Tool for Measuring Internet Path Characteristics," http://www.employees.org/∼bmah/Software/pchar/, June 2000.

[16] A.B. Downey, "Using Pathchar to Estimate Internet Link Characteristics," in *ACM SIGCOMM*, Sept. 1999.

[17] K. Lai and M.Baker, "Measuring Link Bandwidths Using a Deterministic Model of Packet Delay," in *Proceedings ACM SIGCOMM*, Sept. 2000.

[18] "Network Simulator (ns), version 2," http://www-mash.cs.berkeley.edu/ns/.

[19] M. S. Taqqu, W.Willinger, and R.Sherman, "Proof of a Fundamental Result in Self-Similar Traffic Modeling," *ACM Computer Communications Review*, pp. 5–23, Apr. 1997.

[20] K. Thompson, G. J. Miller, and R. Wilder, "Wide-Area Internet Traffic Patterns and Characteristics," *IEEE Network*, pp. 10–23, Nov. 1997.

[21] S. McCreary and K. C. Claffy, "Trends in Wide Area IP Traffic Patterns," Tech. Rep., CAIDA, Feb. 2000.

[22] M. Mathis, *TReno Bulk Transfer Capacity*, Feb. 1999, draft-ietf-ippm-treno-btc-03.txt.

[23] V. Paxson, "Automated Packet Trace Analysis of TCP Implementations," in *Proceedings SIGCOMM Symposium*, Sept. 1997.