Long Distance Visibility of the Ethernet Capture Effect

Bob Melander, Mats Björkman and Per Gunningberg
Dept. of Computer Systems
Uppsala University
Box 325, SE-751 05 Uppsala, Sweden.
E-mail: {Bob.Melander, Mats.Bjorkman, Per.Gunningberg}@docs.uu.se

Abstract— **We present available bandwidth measurements based on packet trains from the Internet along with network simulations that together suggest that the Ethernet capture effect can bias bandwidth estimates. More precisely, the available bandwidth is over-estimated in situations when the capture effect has affected the measurements.**

The capture effect is a short-term unfairness due to the Ethernet CSMA/CD contention algorithm that makes it possible for one host on a LAN to effectively lock out other hosts from gaining access to the medium. The main contribution of this paper is to show that the Ethernet capture effect, which is a phenomenon local to an Ethernet, can be detected in flows even as they have traversed multi-hop WAN paths. The capture effect can therefore disturb and bias end-to-end measurements in the Internet.

I. Introduction

Many network applications (or the protocols they rely on) can benefit from knowledge about properties of the network path they will use. One example of such a property is *available bandwidth*. Intuitively, this is the highest send rate that a sender can use while avoiding to overload the network path. Since the network path is normally shared with traffic from other sources the available bandwidth is dynamically changing. Furthermore, it is an *end-to-end* property since the whole network path is considered. The most congested link of the path is the link that sets the limit on the available bandwidth.

It turns out that measuring and estimating available bandwidth is a non-trivial task. Several methods have been proposed [1], [3], [4] and many potential problems have been identified, e.g. asymmetric routes, multichannel links and limited resolution of clocks [5], [9]. Yet another problem is discussed in [2]. There it is shown that a side effect, called the Ethernet capture effect (ECE) [8], [10], [11], [12], of the Ethernet contention algorithm can make packet train based measurement techniques (e.g. c-probe [3]) *overestimate* the available bandwidth in an *isolated* LAN environment. The core of the problem is that the cross traffic does not interleave with the probe packets properly because of the capture effect.

This paper is a follow up work to [2]. It is based on available bandwidth measurements in the Internet along with simulations performed in the network simulator *ns* [6]. The measurements and simulations suggest that *the ECE is also visible on network paths consisting of several links where the peer networks are Ethernet LANs*. That is, the ECE that is local to one of the peer Ethernets has a long distance effect by disturbing the whole end-to-end measurement. The result is (like in the isolated LAN case) an over-estimation of the available bandwidth.

We also discuss how to avoid the ECE problem in available bandwidth measurements. In particular we describe an alternative probing scheme to packet trains that is resilient to the capture effect.

II. The Ethernet capture effect

The CSMA/CD backoff algorithm of the Ethernet/IEEE 802.3[1] has a side effect that can result in short-term unfairness of access to the medium. This side effect is called the Ethernet Capture Effect (ECE). It essentially means that one host on the LAN has an increased probability of holding on to the channel and sending several consecutive packets even though other hosts are contending for access. The effect is primarily noticable when an Ethernet/IEEE 802.3 LAN is under high load.

A. The Ethernet (802.3) backoff algorithm

The aim of the Ethernet medium access method is to give all stations fair access to the channel in the sense that there are no prioritized stations or classes of traffic.

Whenever a station has a frame to send it checks if the channel is idle and in that case it attempts to transmit the frame. If other stations try to transmit at the same time then all sending stations will detect a collision. To arbitrate between the contending stations the Ethernet/IEEE 802.3 CSMA/CD protocol uses a backoff algorithm where the load offered to the channel plays an important role. When the offered load is high, the stations back off for longer times compared to when the load is low.

This work is supported in part by the SITI/Ericsson CONNECTED project.

[1] We will use the names Ethernet and 802.3 interchangeably.

To estimate the instantaneously offered load, the stations associate a collision counter n with each frame. This counter is initially zero and it is increased by one each time the frame experiences a collision. When a station has detected a collision it uses the collision counter to decide how many slot times, n_s, to wait before attempting to transmit again. This delay, n_s, is chosen as a uniformly chosen random integer in the range $0 \leq n_s \leq 2^k - 1$ where $k = \min(n, 10)$. If the frame experiences 16 consecutive collisions, the retransmission procedure is aborted and the frame is discarded.

The problem with this backoff scheme is that the station that is successful in sending its frame after a collision (i.e., the station that chooses the earliest slot no other station chooses) will start with a new frame having the collision counter set to 0. All other stations involved in the collision will try to retransmit their old frame, and therefore keep their old collision counter values. As a result, if the successful station has another frame to send, it will likely be involved in a new collision with the same stations that were involved in the previous collision. The successful station will choose its random wait time in a more narrow range than the other stations. This will increase its probability of being successful again in the new collision. For every consecutive collision that is won by the same station, the probability for that station to win again (against the same set of competing stations) will increase, and quickly tend towards 1 [10]. This increase of probability leads to the unfairness that is called the Ethernet Capture Effect.

B. Ethernet LANs prone to the capture effect

In order for the capture effect to come into play, network hosts that are interconnected using Ethernet interfaces must share the same collision domain. That is, without collisions, no capture effect. In old style (and mostly outdated) Ethernets where hosts are connected using coax-cables, all hosts belong to the same collision domain. This is also the case when the hosts are interconnected using twisted pair cables and hubs, since a hub works as a broadcast medium.

Today, most Ethernets (at least in commercial use) are built around switches where the hosts are interconnected point-to-point. This means that the hosts no longer share one common collision domain. However, in some switches it is still possible for hosts attached to the switch to share collision domains. It happens in the following case. If a set of hosts (attached to the switch) simultaneously send packets to another host (also attached to the switch) and the total incoming traffic exceeds the link capacity of the outgoing link, packets will be queued. If the overload continues, the queue will overflow. Some switches signal collision in those situations to trigger a rate reduction of the incoming flows. A concrete real-world example of a LAN host that other hosts repeatedly try to flow traffic through is the default gateway (the router) for the LAN. In all fairness it should be pointed out that queue overflow in Ethernet switches is not common because the queue buffer space is typically large.

III. Measurement and simulation study

The measurements that we have performed include hosts at Uppsala universitet (Sweden), University of Massachussets at Amherst (USA), Cambridge university (England), Freie Universität Berlin (Germany) and University of New South Wales (Australia). The LAN hosts have been interconnected using a hub to ensure that they are in the same collision domain.

In what follows, we will present results from the measurements between Uppsala and Amherst. The measurements to the other locations corroborate these results. The network path the packets traversed from the Uppsala (802.3) LAN to the Amherst (802.3) LAN during the measurements consisted of a Gigabit Ethernet link, a couple of OC-3 and OC-48 links, an OC-12 trans-atlantic link, another couple of OC-3 and OC-48 links, and a DS-3. The LAN capacities were 10 Mbps. The round-trip time was typically in the vicinity of 175 ms.

It is difficult to draw conclusions solely based on the measurements in the real Internet since the network environment cannot be fully controlled. To verify our hypotheses regarding the capture effect we have therefore also performed simulations using the network simulator *ns*. The idea is; if the results from those simulations show the same characteristics as those from the real measurements, that is a strong indication that our hypotheses are valid.

To make the discussion and argumentation easier to follow, the real measurements and the simulations have been grouped into a set of experiments. Each experiment illustrates a certain scenario and shows if/how the Ethernet capture effect comes into play. In the first experiments, 1a and 1b, all measurements are done locally on a loaded Ethernet LAN. Experiment 2 considers a long distance measurement (i.e. over a WAN) where the local LAN is loaded (and the remote LAN is not). In experiment 3, the load on both LANs is neglectible whereas there is congestion in the WAN. The fourth experiment is the combination of experiment 2 and 3, i.e. the local Ethernet is loaded (whereas the remote LAN is not) and there is congestion in the WAN). The final experiment considers a scenario where the remote LAN is loaded (and the local LAN is not) while there is congestion in the WAN.

A. Simulation topology

Figure 1 shows the topology used in the simulations. The nodes marked R correspond to routers and the shaded nodes

correspond to hosts generating cross traffic (i.e. flows competing with our probe flows). Each shaded node actually represents several traffic generating hosts. The dashed lines with arrows are the cross traffic flows. The white nodes on the LANs correspond to the probe source host and the probe destination host.

We have tried to mimic the real path as much as possible but to reduce simulation time and memory use we have limited the WAN link capacities to that of an OC-3, i.e. 155 Mbps. In addition, while the real path consists of 12 WAN hops, the simulation topology consists of 7 WAN hops. This reduction of hops and capacity does not have any significant impact on the interpretation and validity of the results. The LAN capacities are 10 Mbps. The WAN bottleneck, i.e. the WAN link with the least link bandwidth, has DS-3 capacity, i.e. 45 Mbps. In simulations where there is congestion in the WAN, that happens on the WAN bottleneck.

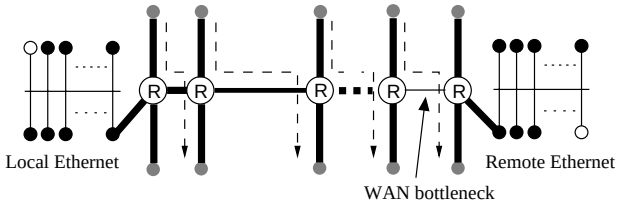

Fig. 1. The topology used in the simulations.

The cross traffic consist of a mixture of flows where the packets are sent either according to a Poisson process or with a uniformly distributed random inter-packet spacing. The packet size is varied uniformly from 50 bytes to the size of an entire Ethernet frame, with a mean size of 500 bytes. This is somewhat simplistic and does not perfectly reflect the traffic in the real Internet. Nonetheless, we believe that it is realistic enough from the perspective of this study.

B. The measurement and estimation procedure

All real and simulation measurements have been performed similarly. A probe source (on the local LAN) sends 60 trains of probe packets to the probe destination (on the remote LAN). A probe train consists of 100 packets and the packets are sent back-to-back, i.e. at the link speed of the Ethernet. Each probe packet is sized to fill an entire Ethernet frame. Upon reception of a probe packet, the probe destination host record the arrival time. Hence, for each train i, $0 \leq i \leq 59$, there will be 100 time stamps, $t_{i0}, t_{i2}, \ldots, t_{i99}$.

When the probe packets traverse the path from sender to receiver they will be interleaved by cross traffic packets. The resulting separation will be manifested in the time stamps.[2] The bandwidth estimate is calculated as the amount of data sent divided by the separation in time (between the arrival of the first and last bits of the data). That is, from the time stamps collected for train i, bandwidth estimates can be calculated for different train lengths j, $1 \leq j \leq 99$, as $bw_i(j) = \frac{j \cdot s}{t_{ij} - t_{i0}}$ where s is the size of a probe packet.

However, instead of performing the calculation above on individual trains we perform it on the mean time separation, i.e. $bw(j) = \frac{j \cdot s}{\Delta t_j}$ where $\Delta t_j = \frac{1}{59} \sum_{i=0}^{59} t_{ij} - t_{i0}$. This makes it easier to discover persistent patterns or artifacts in the measurements. In the graphs presented in subsequent sections, $bw(j)$ is plotted as a function of j, i.e. train length, unless otherwise stated.

C. Experiment 1a

In the first experiment all measurements are done locally on one Ethernet. That is, the probe source host and probe destination host reside on the same LAN. During the probing session one other host is generating 5 Mbps of cross traffic. The available bandwidth (if defined as the currently unused portion of the link capacity) is $10 - 5$ Mbps = 5 Mbps. This is a setup identical to the one used in [2].

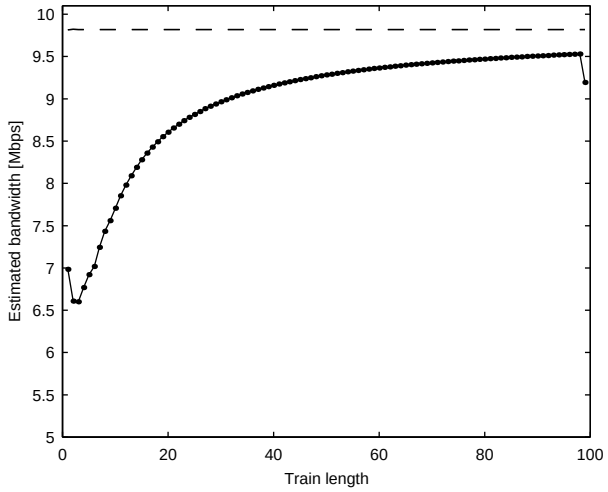

Fig. 2. Simulation. One LAN host probing at 10 Mbps while another is generating traffic at 5 Mbps.

In figure 2, which is from the simulation measurement, the probe rate is plotted with the dashed line and the estimated bandwidth is plotted with the line with dots. The plotted probe rate is slightly less than the nominal rate 10 Mbps because all calculations are done using packet payload size ignoring the size of headers. Figure 3 shows a similar graph but for the real measurement (albeit without the probe rate plotted). As can be seen, the

[2]We ignore here the fact that packets separated in time at one router queue can be pushed together again in the queues of subsequent routers.

available bandwidth (5 Mbps given the same definition as earlier) is grossly over-estimated by the probe train method used here. The variance in the estimated bandwidths in these measurements is roughly 1%.

In both plots there is an initial dip in the estimated bandwidth curve which then rises towards 9.5 Mbps. This behavior can be explained in terms of the Ethernet capture effect. The key is that a host that has been successful in a contention situation stands a better chance of winning a subsequent contention situation than the unsuccessful hosts.

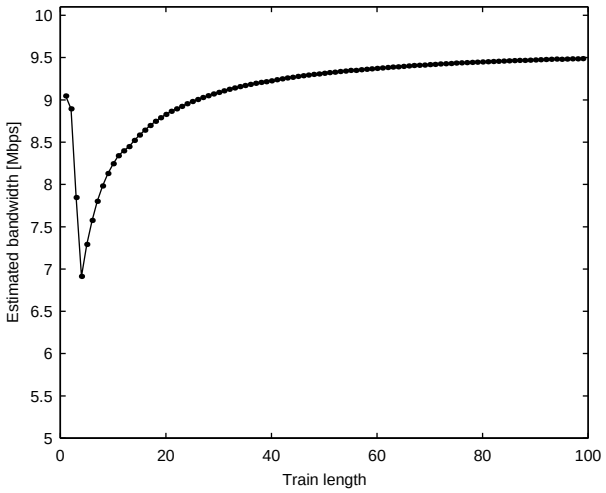

Fig. 3. Real measurement. One LAN host probing at 10 Mbps while another is generating traffic at 5 Mbps.

As long as there is only the 5 Mbps of cross traffic, there is enough capacity to send the packets without queues building up at the sender. Then, when a probe train is about to be sent, the total traffic (i.e. cross traffic and probe traffic) exceeds the LAN capacity. Since the cross traffic only uses half of the total LAN capacity there is a reasonably good chance that the probe host will find the medium non-busy. It can then start to send probe packets back-to-back. During the time when probe packets are being sent, cross traffic packets will be queued at their sender. The shorter the probe train, the shorter the time during which a cross traffic queue will build up and consequently, the shorter the cross traffic queue. Furthermore, with shorter probe trains, there are fewer possibilities of collisions (one for each probe packet).

Consequently, a fair amount of short trains will be successfully sent back-to-back. Even if there are collisions, the difference in probability of successfully winning a contention between the probe host and the cross traffic host cannot grow very large (again since the number of possible collisions is limited). Hence, the contention situations that are lost will not be many enough to lower the average value.

As the probe trains get longer, the time for the cross traffic to build a queue increases. This longer cross traffic queue translates into the cross traffic host being a more aggressive combatant in subsequent contention situations. Consequently, fever probe packets will be sent back-to-back. This in turn, results in a lower bandwidth estimate and thus the downhill side of the dip.

Increasing the train length further will give the cross traffic even longer times to build a queue. Like before this will make it more aggressive. With the same reasoning as above that should lower the bandwidth estimate even further. However, as shown by the graphs there is something counteracting that. That "something" is the fact that while the cross traffic host transmits, the probe host builds a queue (just as the opposite happens when the probe host transmits). Since the probe rate is higher than the cross traffic rate, the probe queue builds faster than the cross traffic queue. That together with the increased number of collision possibilities as the train gets longer effectively makes the probe traffic push the cross traffic away. This attributes to the uphill part of the dip.

Figure 4 illustrates the capture effect in an alternative way. The bandwidth estimate is now calculated as $bw(j) = \frac{s}{\Delta' t_j}$ where $1 \leq j \leq 99$ as before and $\Delta' t_j = t_{j+1} - t_j$. Hence, the estimate is calculated from the time separation between consecutive packets. As can be seen in the graph, the probe pairs (except for the first one) are essentially less and less interleaved by cross traffic packets. After the 11th packet pair no cross traffic packets interleave with the probe packets. That is, those probe pairs are all sent back-to-back.

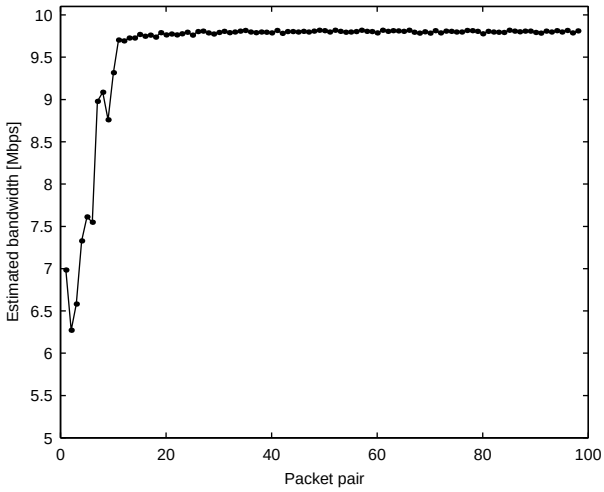

Fig. 4. Simulation. The bursts of probe packets when one LAN host is probing at 10 Mbps while another is generating traffic at 5 Mbps.

We have noticed that the dip in the graphs for the simulation measurements is smaller than the dip in the graphs for the real measurements. We have not been able to explain this.

D. Experiment 1b

We now change the setup slightly so that there are five cross traffic hosts on the LAN instead of one. Each of them generates traffic at a rate of 1 Mbps so the total load from the aggregated cross traffic is 5 Mbps as in Experiment 1a.

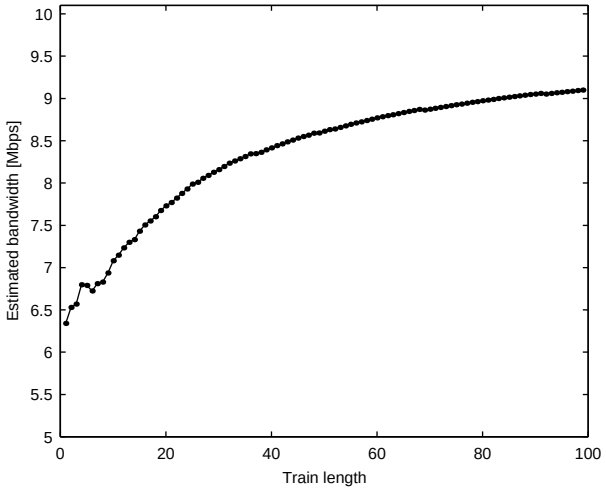

Fig. 5. Simulation. One LAN host probing at 10 Mbps while five others are generating traffic at 1 Mbps each.

Figure 5 shows a plot of the estimated bandwidth from a simulation measurement. A real measurement will yield a similar result, see [2] for such an experiment. The curve has the same characteristics as the curves in Experiment 1a, except that there is no downhill side of the dip. Another difference is that the curve rises more slowly and only manages to reach 9.1 Mbps for the longest trains compared to 9.5 Mbps in Experiment 1.

This is what would be expected. Even though the *total* load from the cross traffic has not changed, the rate for each individual cross traffic flow is only $\frac{1}{5}$ of what it used to be. The probe rate is unchanged. This means that the probability that the probe host will be involved in a collision is roughly the same. However, there are now one or several of the five cross traffic hosts that can be involved in the collision. This effectively reduces the difference in probability of winning subsequent contention situations between the probe host and the cross traffic hosts. That increases the likelihood of cross traffic packets interleaving with probe packets which has a lowering effect on the curve.

Even with the five cross traffic hosts in this scenario the available bandwidth is significantly over-estimated by the packet train method. Not even short trains give reasonable estimates.

E. Experiment 2

In the remaining experiments (with the exception of experiment 5b) the probe destination host reside on the remote (Amherst) LAN. Hence, the probe packets will always traverse the WAN network path on their way from source to destination. Furthermore, when there is significant cross traffic on the LANs, that traffic is always generated by one host.

The scenario we have tried to achieve in this experiment is to have a low load on the remote LAN and on the WAN links while there is 5 Mbps of cross traffic on the local LAN. For that reason, the real measurements were performed at 8.30am in the morning Swedish time. This means that the time in Amherst was 2.30am. Our assumption was that at this time the traffic in the WAN would be modest. In particular, it was known that the 45 Mbps WAN bottleneck was not congested.

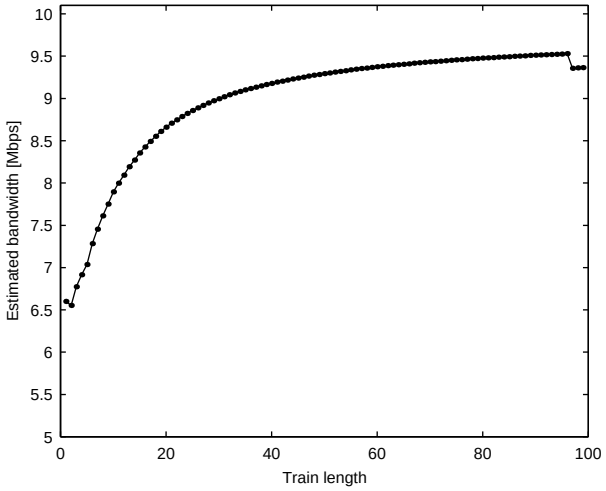

Fig. 6. Simulation. One LAN host probing at 10 Mbps to host at remote LAN. Period of low cross traffic in LANs and WAN.

Figures 6 and 7 contain plots of the estimated bandwidth from the simulated measurement and the real measurement. They show a striking similarity with their counterparts in Experiment 1. This is not very surprising given the assumption of no congestion in the WAN and insignificant cross traffic on remote LAN. The impact of the limited cross traffic on the path is so small that it can essentially not be observed in the bandwidth estimates. As a result, the capture effect on the sender LAN is clearly visible on the receiver LAN. The variance in these measurements (as well as in the measurements in experiments 3 and 4) is about 1.5%.

The periodic shifts that appear for train lengths 12 and larger in the real measurement plot (Figure 7) are most likely caused by the probe destination host. It turned out that during the measurement, a Netscape process was running on the probe destination host at close to 100% cpu usage. Since the time stamping is done at user space, process scheduling together with the eager Netscape process probably disturbed the time stamps. The effect is limited though and does not overshadow the main characteristics of the curve.

As with the measurements on the local LAN, the available

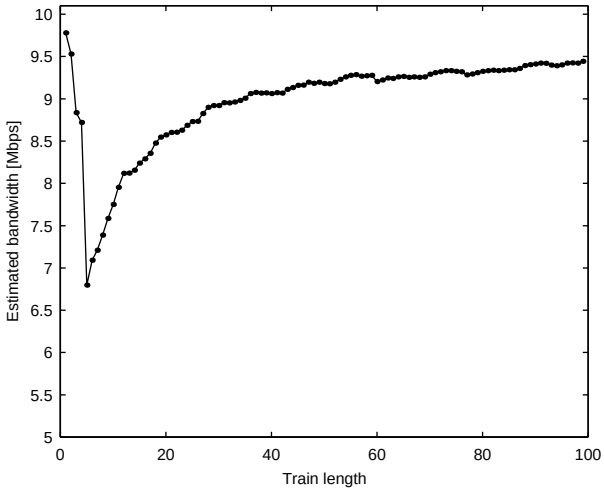

Fig. 7. Real measurement. One LAN host probing at 10 Mbps to host at remote LAN. Period of low cross traffic in LANs and WAN.

bandwidth is also here over-estimated significantly.

F. Experiment 3

We now consider a scenario where there is congestion in the WAN[3] while there is neglectible cross traffic in the LANs. In this experiment, the real measurement was performed at 8.30pm Swedish time, i.e. 2.30pm Amherst time. At that time it is reasonable to assume that there is reasonable amounts of cross traffic in the WAN. In particular, the load on the 45 Mbps WAN bottleneck link was approximately 42 Mbps.

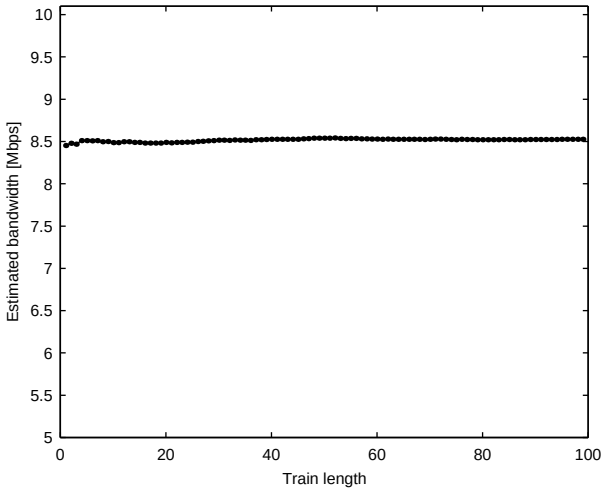

Fig. 8. Simulation. One LAN host probing at 10 Mbps to host at remote LAN. Period of low cross traffic in both sender and receiver LAN and 42 Mbps of cross traffic on WAN bottleneck (link capacity 45 Mbps).

The estimated bandwidths from the simulation and the real measurement are plotted in figure 8 and 9, respectively. As can be seen the estimated bandwidths are approximately 8.5 Mbps.

[3]More precisely, congestion happens when the probe traffic is added

All routers in the simulation schedule packets *first-come-first-serve*. According to [7], the relationship between a flow's outgoing rate r_o and it's incoming rate r_i at a router is then given by

$$r_o = \begin{cases} r_i & \text{if } r_i \leq l - x \\ \frac{r_i}{x + r_i} l & \text{if } r_i > l - x \end{cases} \quad (1)$$

where x is the rate of the aggregated cross traffic and l is the capacity of the outgoing link.

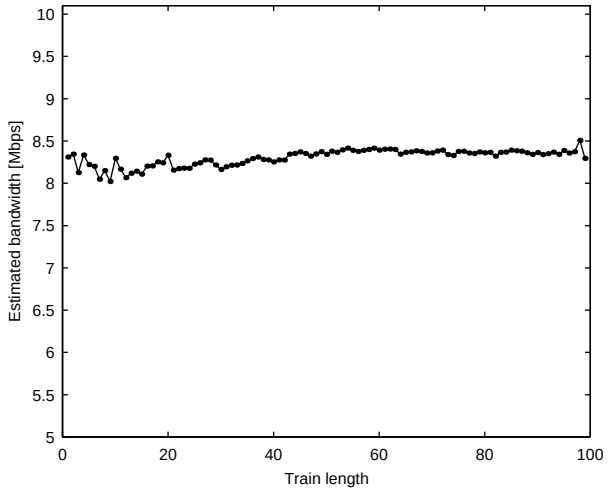

Fig. 9. Real measurement. One LAN host probing at 10 Mbps to host at remote LAN. Period of low cross traffic in both sender and receiver LAN and approximately 42 Mbps of cross traffic on WAN bottleneck (link capacity 45 Mbps).

Given that the congested link is the WAN bottleneck, the expected estimated bandwidth should be $\frac{9.8}{42+9.8} \cdot 45$ Mbps = 8.51 Mbps (where we have used the second part of equation 1 since $9.8 > 45 - 42 = 3$). This is very to close to the bandwidth estimate in the simulation measurement.

Since the bandwidth estimate from the real measurement is also close to 8.51 Mbps and the load on the WAN bottleneck was approximately 42 Mbps, it is reasonable to guess that the bottleneck router also schedules packets first-come-first-serve. The discrepancy could be due to the fact that the traffic flows are discrete (i.e. quantisized into packets) while the equation assumes continuous flows. It is finally worth pointing out, as observed in [7], that the available bandwidth (3 Mbps = 45 - 42 Mbps) is over-estimated in this case too. The difference is that the Ethernet capture effect is not the reason here.

G. Experiment 4

In this experiment the scenario is still that there is congestion in the WAN (just as in Experiment 3 with an approximate 42 Mbps load on the 45 Mbps WAN bottleneck). In addition,

there is also 5 Mbps of cross traffic on the local LAN whereas the cross traffic on the remote LAN is neglectible.

There are consequently two phenomena that will interact in this case, the capture effect on the local LAN and the incoming flow/outgoing flow relationship as dictated by equation 1. The question is what the net effect will be. The plots of the estimated bandwidths from the simulation measurement and the real measurement shown in the figures 10 and 11, respectively, hold the answer.

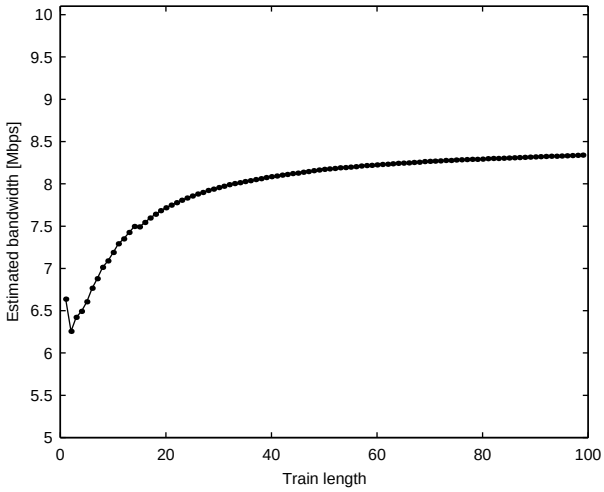

Fig. 10. Simulation. One LAN host probing at 10 Mbps to host at remote LAN. One host at sender LAN generating cross traffic at 5 Mbps. Period of low cross traffic in remote LAN and 42 Mbps of cross traffic on WAN bottleneck (link capacity 45 Mbps).

The main characteristics of the curves are still the same. The curves begin with a dip for short trains followed by an increase that is weakened as the train length increases. A closer look at the scales on the y-axis reveals that the curves are more compact than before though. The dips are deeper (6.3 Mbps and 5.6 Mbps vs. 6.5 Mbps and 6.7 Mbps in figures 6 and 7)) and the final value is lower (approximately 8.4 Mbps vs. approximately 9.5 Mbps from the same graphs in experiment 2).

This can be explained in the following way (where we consider the simulation measurement). The probe packets are generated back-to-back at 10 Mbps on the local LAN. Because of the 5 Mbps of cross-traffic, the capture effect comes into play. The rate of the probe flow as it leaves the local LAN will therefore be as in figure 2. The probe packets then start to traverse the WAN. The effect of the cross traffic on the first WAN links is limited since there is enough capacity to handle to total load from the cross traffic flows and the probe flow.

Eventually when the probe packets reach the WAN bottleneck there is congestion and the second part of equation 1 comes into effect. That should push the end of the curve (i.e. for train lengths close to 100) towards 8.3 Mbps (since $r_i = 9.5$ Mbps

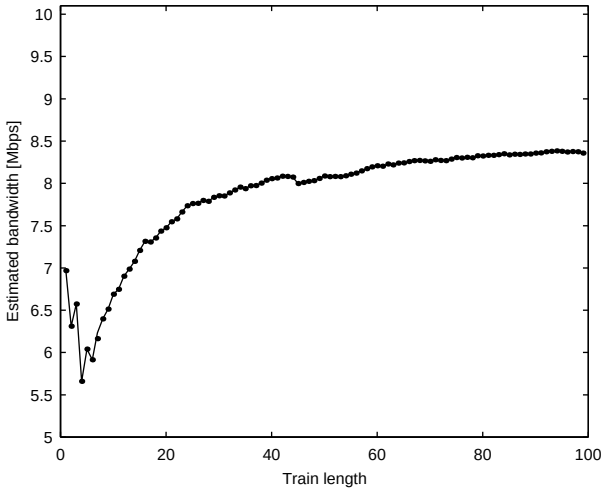

Fig. 11. Real measurement. One LAN host probing at 10 Mbps to host at remote LAN. One host at sender LAN generating cross traffic at 5 Mbps. Period of low cross traffic in remote LAN and approximately 42 Mbps of cross traffic on WAN bottleneck (link capacity 45 Mbps).

and $9.5/(42 + 9.5) \cdot 45 = 8.3$). The graph shows that this indeed happens. Likewise, for train lengths equal to 20, the curve should come close to 7.57 Mbps (since according to figure 2, $r_i \approx 8.5$ Mbps and $8.5/(42 + 8.5) \cdot 45 = 7.57$). Again, we see a strong agreement in values. For shorter probe trains, the probe rate is close to or slightly under the rate at which the first part of equation 1 should be used. That makes it harder to see a strong conformance to the equation.

As in the case with a non-congested WAN, the Ethernet capture effect on the sender LAN has apparently affected the entire long distance measurement. We can also see that the available bandwidth is again over-estimated but to a lesser degree than before (especially for shorter trains). This is because of the combined actions of the Ethernet capture effect and the first-come-first serve scheduling at the bottleneck router. Still, the over-estimation is significant.

H. Experiment 5a

We now consider a scenario opposite to the one in experiment 4. More specifically, there is approximately 42 Mbps of cross traffic on the WAN bottleneck, which will then be congested by the probe traffic. At the same time, there is neglectible load on the local LAN whereas the load on the remote LAN is 5 Mbps. Because of this, there is reason to suspect that the Ethernet capture effect will again interact with the incoming flow/outgoing flow relationship given by equation 1. However, the order of the locations where the two phenomena happen is now reversed compared to the one in experiment 4. It is therefore reasonable to assume that the net effect may be different.

Figure 12 contains a plot of the estimated bandwidth from the

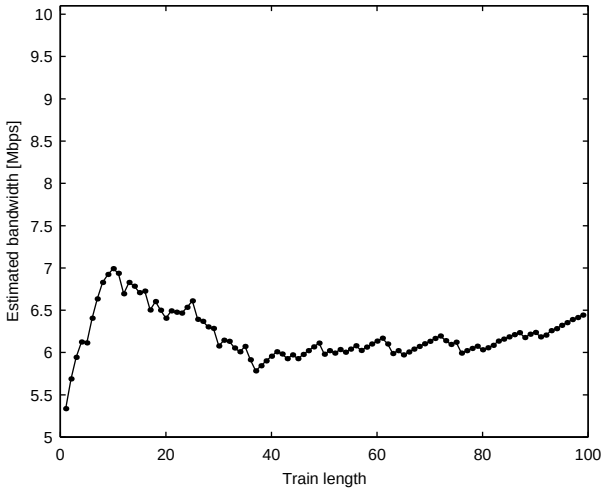

Fig. 12. Simulation. One LAN host probing at 10 Mbps to host at remote LAN. Period of low cross traffic on local LAN. One host at remote LAN generating cross traffic at 5 Mbps. The load from cross traffic on the WAN bottleneck (link capacity 45 Mbps) is 42 Mbps.

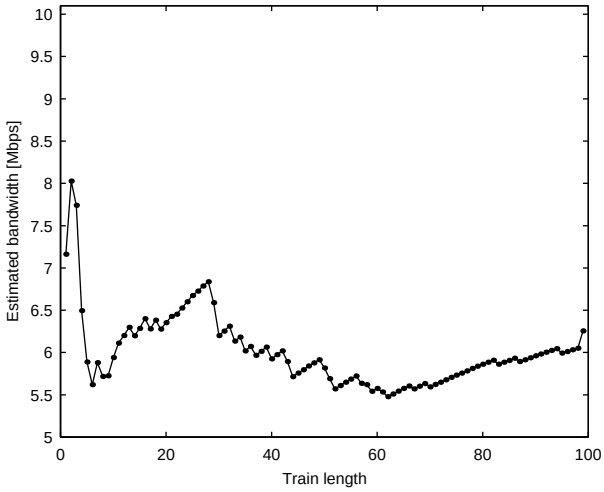

Fig. 13. Real measurement. One LAN host probing at 10 Mbps to host at remote LAN. Period of low cross traffic on local LAN. One host at remote LAN generating cross traffic at 5 Mbps. The load from cross traffic on the WAN bottleneck (link capacity 45 Mbps) is approximately 42 Mbps.

simulation measurement. A similar graph for the real measurement is found in figure 13. The variance in these measurements is approximately 7%, which is somewhat higher than the measurements in the earlier experiments. It is obvious that these graphs look fundamentally different from those in experiments 1, 2, and 4. If studied more carefully though, it can be noticed that the graphs in figures 12 and 13 themselves have similarities (except for the initial part, i.e. for train lengths 6-7 and less). There is a hill in both curves that spans over train lengths 1 to 40 in the simulation measurement and train lengths 9 to 60 in the real measurement. Those hills are then then followed by weak increases in estimated bandwidth that continues up to the maximum train length.

In order to explain this behavior let us follow the flow of probe packets as they traverse the network. Since there is neglectible cross traffic on the local LAN the probe packets will enter the first WAN link separated in time dictated by the link speed of the LAN. As they proceed in the WAN, the probe packets will only suffer from minor disturbances from the cross traffic there. This is because there is on average enough capacity to handle both the probe flow and the cross traffic flows. Eventually the probe packets reach the WAN bottleneck link, which gets congested. Upon leaving the bottleneck link, the rate of the probe flow will essentially be 8.5 Mbps as shown in figure 8 (for the same reasons as in experiment 3). This is the rate of the probe flow as it arrives at the remote LAN.

Since the total load on the remote LAN is 13.5 Mbps ($5 + 8.5$ Mbps) which exceeds the link capacity, queues will build up on the LAN hosts. This is the trigger for the capture effect. However, compared to the previous experiments where the measurements were affected by the capture effect, the rate of probe flow rate is now lower (8.5 Mbps instead of 9.8 Mbps). This means that while the cross traffic host is transmitting, the queue of probe packets will not grow as fast as before. In that sense, the difference in aggressiveness between the probe flow and cross traffic flow has been reduced. That together with the lower total load (which reduces the risk of collisions) decreases the difference in probability of winning contention situations between the cross traffic host and the host forwarding the probe flow. Hence, the chances that the cross traffic will interleave with the probe packets is increased. That is what attributes to the general lowering of the estimated bandwidth curves.

A direct consequence of the reduced difference in probability is that it makes it harder to predict the exact look of the estimated bandwidth curve (since neither of the LAN hosts is as dominant as when there is a big difference in probability). Still, there is an explanation of the observed hills in the estimated bandwidth plots. The uphill side correspond to times when the probe flow queue has grown and the capture effect has kicked in and is working to the advantage of the probe flow. The reduced rate of the probe flow implies that, for short train lengths, the queue on the host forwarding the probe flow eventually drains completely. When that happens (which varies among different probe trains due to randomness), the cross traffic host gets access to the medium (and resets it collision counter whereby the difference in probability is temporarily zero again). That attributes to the downhill side.

During the time when the cross traffic is transmitting, the packets in the probe flow will be queued. Eventually the cross traffic will have emptied its queue of packets (that was built

up during the time when probe packets were transmitted), thus granting the probe flow access to the medium. Alternatively, there will eventually be a contention situation that the cross traffic host will lose. That also gives the probe flow access to the medium. Now, since there is a persistent 3.5 Mbps ($8.5 + 5 - 10$ Mbps) over-load, the probe queue will on average grow larger and larger the further into a probe session (or equivalently, the further into a probe train) we get. This is what gives the slow increase in the estimated bandwidth that is visible after the first hill.

I. Experiment 5b

As a comparison with the measurements in example 5a, we now consider again a single isolated LAN with 5 Mbps of cross traffic generated by one host. The probe source and destination hosts reside on the same local LAN and the probe packets will not traverse a WAN. That is, the same scenario as in example 1. A difference in this case though, is that the probe packets are not sent back-to-back (i.e. at link speed, 10 Mbps). Instead, the probe flow rate is 8.3 Mbps.

This should basically be equivalent to the what happens in experiment 5a except that the probe packets there have been affected by cross traffic so the separation of the probe packets may not be as regular as now. The rate of the probe flow when arriving at the remote LAN in experiment 5a is also slightly higher (on average 8.5 Mbps vs. 8.3 Mbps here).

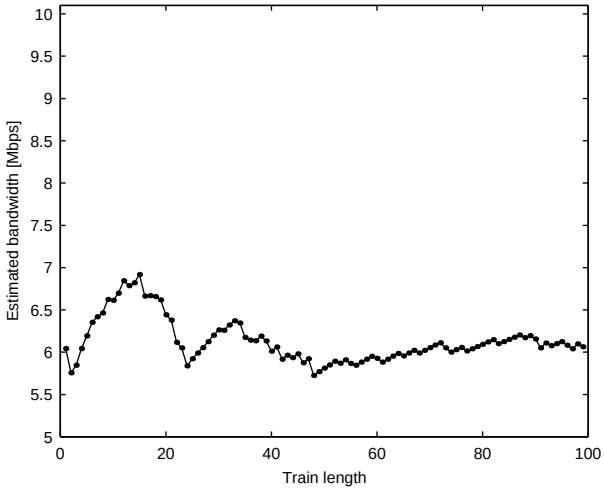

Fig. 14. Simulation. One LAN host probing at 8.3 Mbps to host at remote LAN. Period of low cross traffic on local LAN and on WAN. One host at sender LAN generating cross traffic at 5 Mbps.

Figure 14 shows a graph of the estimated bandwidth from a simulation measurement. The curve has indeed similarities with the curve in figure 12. The "level" of the curves is approximately the same (6 Mbps) and, starting at train length 50, we see the weak increase in estimated bandwidth as the train length increases. There is one additional hill for trains shorter than 50 in the new plot. This is probably due to the somewhat lower rate of the probe flow (possibly in conjunction with the reduced difference in probability to win a contention).

IV. Avoiding the Ethernet capture effect

The graphs in Experiment 5a and 5b give some insight to how probing should be done in order to avoid to trigger the Ethernet capture effect. The problem with probe trains (where packets are sent back-to-back as illustrated in figure 15*a*) is that in a LAN overload situation, the train of probe packets will be queued. The longer the queue, the more possible it becomes for the probe flow to, especially in cases with few cross traffic hosts, monopolize the medium.

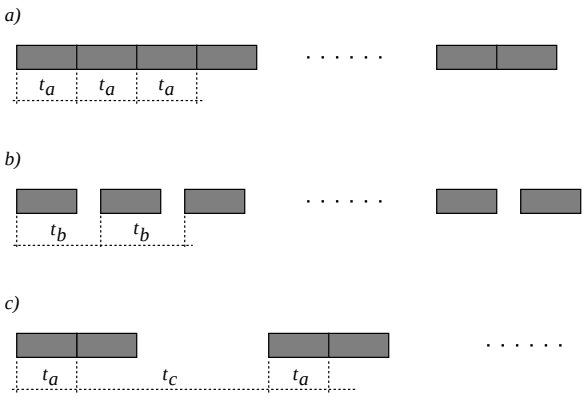

Fig. 15. Three different probing patterns, a) probe train with packets sent back-to-back (time separation equal to t_a), b) probe trains with packets separated with time $t_b > t_a$, and c) trains of probe pairs sent back-to-back where the pairs are separated in time by t_c. The times t_b and t_c should be large enough to reduce risk of queue build-up.

Thus, in order to avoid triggering the capture effect, the probe packets should not be sent as a back-to-back train. One alternative could be to increase the spacing between all the packets (as shown in figure 15*b*). This would correspond to the measurements in Experiment 5a and 5b. However, that would reduce the minimum bandwidth the probe train method used here can measure. The reason is that the probe rate sets a lower limit on the bandwidth that can be measured (since with gaps between the probe packets, the cross traffic packets may interleave the probe packets unnoticed). Even more serious, in a situation where the load from the cross traffic is high and the probe rate is low (i.e. when $t_b >> t_a$), the capture effect will work in favor of the cross traffic. This would lead again lead to a biased bandwidth estimate (because of improper interleaving of probe packets and cross traffic packets).

A better approach is instead to send a limited number of probe

packets back-to-back, e.g. 2, instead of a whole train. That will make it impossible for the probe flow queue to grow large thereby avoiding to trigger the capture effect. One probe pair alone though is most likely not enough to get a statistically good value. This is especially the case if the cross traffic is bursty, since the probe pair may fall in between two bursts or be affected by one long burst, both cases leading to a biased value.

By sending several well separated probe pairs where the probe packets in a pair are sent back-to-back, the number of samples can be made large enough to get good statistical accuracy while at the same time minimize the impact of the Ethernet capture effect. This is illustrated in figure 15*c*. It is important that t_c is made large enough so that a queue of probe packets does not grow large. At the same time, t_c must not be too large since the cross traffic intensity may then change during the (prolonged) probe session. Hence, there is clearly a trade-off here.

The idea of sending trains of packet pairs is used in the TOPP available bandwidth measurement method, which is described and discussed in [7]. A difference there is that the packets in the well-separated pairs are not only sent back-to-back but with a varying separation. That together with analysis based on equation 1 is used to form the bandwidth estimates.

V. Conclusions

The measurements performed in the real Internet coupled with the simulation study has led us to draw the following conclusions:

- The Ethernet capture effect, while local to a LAN, can affect traffic flows to an extent that the affect is visible even after the flow has traversed a multi-hop (congested or non-congested) WAN.
- Train based methods, such as the one used in this paper, that attempt to measure available bandwidth fail to give accurate estimates when either end of the probe path is a loaded Ethernet. The available bandwidth is typically over-estimated in such cases.
- To avoid triggering the Ethernet capture effect, probe packets should not be sent back-to-back in long trains but preferably as well separated pairs of packets.

Finally, it is worth noting that the train based method used in this paper failed to give good estimates of available bandwidth even when the measurements were not affected by the Ethernet capture effect. The reason for this failure is that the routers use first-come-first-serve packet scheduling.

An issue that has not been discussed in this paper is how TCP is affected by the ECE. Ramakrishnan and Yang has studied this in [10] where they also propose a slightly different Ethernet back-off algorithm that is supposed to overcome the capture effect.

VI. Acknowledgements

Thanks to Anders Andersson at DoCS, Uppsala University, for helping out with the necessary network configuration changes at Uppsala to make the measurements possible. We are also grateful to the people and network administrators at the remote network sites for supplying us with user accounts and suitable machine setups.

References

[1] Mark Allman and Vern Paxson. On Estimating End-to-End Network Path Properties. In *SIGCOMM '99 Conference Proceedings*, pages 263–273, Cambridge, MA, USA, August 31–September 3, 1999. ACM SIGCOMM Computer Communication Review, 29(4).

[2] Mats Björkman and Bob Melander. Impact of the Ethernet Capture Effect on Bandwidth Measurements. In *Networking 2000 Conference Proceedings*, pages 156–167, Paris, France, May 2000.

[3] Robert L. Carter and Mark E. Crovella. Measuring bottleneck link speed in packet-switched networks. Technical Report TR-96-006, Boston University Computer Science Department, Boston, MA, USA, March 1996.

[4] Srinivasan Keshav. A control-theoretic approach to flow control. In *SIGCOMM '91 Conference Proceedings*, pages 3–15, Zürich, Switzerland, September 3–6, 1991. ACM SIGCOMM Computer Communication Review, 21(4).

[5] Kevin Lai and Mary Baker. Measuring bandwidth. In *Proceedings of IEEE INFOCOM '99*, New York, USA, March 21-25, 1999.

[6] Steven McCanne, Sally Floyd, Kevin Fall, Kannan Varadhan et al. Network simulator - ns2. http://www-mash.cs.berkeley.edu/ns/, 1997

[7] Bob Melander, Mats Björkman and Per Gunningberg. A New End-to-End Probing and Analysis Method for Estimating Bandwidth Bottlenecks. In *Global Internet 2000 Conference Proceedings*, San Francisco, CA, USA, November 2000.

[8] Mart L. Molle. A New Binary Logarithmic Arbitration Method for Ethernet. Technical Report CSRI-298, Computer Systems Research Institute, University of Toronto, Toronto, Canada, April 1994.

[9] Vern Paxson. End-to-end internet packet dynamics. In *SIGCOMM '97 Conference Proceedings*, pages 139–152, Cannes, France, September 14–18, 1997. ACM SIGCOMM Computer Communication Review, 27(4).

[10] K. K. Ramakrishnan and Henry Yang. The Ethernet Capture Effect: Analysis and Solution. In *Proceedings of IEEE 19th Conference on Local Computer Networks*, MN, USA, October, 1994.

[11] Rich Seifert. The Effect of Ethernet Behaviour on Networks using High-Performance Workstationss and Servers. Technical Report, Networks and Communications Consulting, Cupertino, CA, USA, March 1995.

[12] Brian Whetten, Stephen Steinberg and Domenico Ferrari. The Packet Starvation Effect in CSMA/CD LANs and a Solution. University of California at Berkeley.