Speed Testing without Speed Tests: Estimating Achievable Download Speed from Passive Measurements

Alexandre Gerber, Jeffrey Pang, Oliver Spatscheck, Shobha Venkataraman
AT&T Labs - Research
180 Park Ave.
Florham Park, NJ
{gerber,jeffpang,spatsch,shvenk}@research.att.com

ABSTRACT

How fast is the network? The speed at which real users can download content at different locations and at different times is an important metric for service providers. Knowledge of this speed helps determine where to provision more capacity and helps detect network problems. However, most network-level estimates of these speeds today are obtained using active "speed tests" that place substantial load on the network and are not necessarily representative of actual user experiences due to limited vantage points. These problems are exacerbated in wireless networks where the physical locations of users play an important role in performance. To redress these problems, this paper presents a new technique to estimate achievable download speed using only flow records collected passively. Estimating achievable speed passively is non-trivial because the measured throughput of real flows is often not comparable to the achievable steady-state TCP rate. This can be because, for example, flows are small and never exit TCP slow start or are rate-limited by the content-provider. Our technique addresses these issues by constructing a *Throughput Index*, a list of flow types that accurately estimate achievable speed. We show that our technique estimates achievable speed more accurately than other techniques in a large 3G wireless network.

Categories and Subject Descriptors

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

General Terms

Measurement

Keywords

throughput, passive, measurement, UMTS, 3G, wireless

IMC'10, November 1–3, 2010, Melbourne, Australia.
Copyright 2010 ACM 978-1-4503-0057-5/10/11 ...$10.00.

1. INTRODUCTION

The achievable throughput of a steady-state TCP flow at a given time and location in a network, hereafter referred to as *max-throughput*, has long been recognized as an important metric for service providers.[1] By measuring max-throughput a service provider can determine locations where provisioning more capacity would benefit users and detect temporary network problems. The increasing popularity of multimedia downloads and streaming only magnifies the importance of measuring max-throughput. Yet, the rapid shift from wired to wireless access links makes max-throughput more difficult to estimate using traditional active measurements. In this paper, we develop a new technique to estimate max-throughput passively. We show that our approach estimates max-throughput more accurately than other techniques in a large 3G wireless network.

A number of active probing techniques exist to estimate available capacity, but they are often insufficient to estimate max-throughput. For example, researchers have proposed lightweight active probing techniques based on measuring the inter-packet spacing between pairs of packets (e.g., [3, 5, 8]). These techniques do not work well in wireless networks because packets are often delayed for reasons other than congestion (e.g., due to physical layer loss or due to scheduling into Transmission Time Intervals of 10 ms or more [7]). Moreover, these techniques do not measure the final fair-share and loss-induced throughput that a TCP flow would achieve. Thus, the state-of-the-art for measuring max-throughput in such networks is to periodically download large files from a number of active probes while measuring their achieved throughput. While such experiments precisely characterize the max-throughput to each active probe, they have three obvious limitations: First, they place additional load on the network; second, they are expensive to deploy and maintain; and third, they do not capture the experience of real users at different vantage points.

Due to these limitations, active probes cannot scale to cover a representative portion of an entire wireless network. This paper examines whether passive measurements of real flows within a network are sufficient to estimate max-throughput. Estimating network metrics passively is not new: they have been used to determine bottleneck link capacities [9], to determine RTTs [12], and to find bottlenecks in a 3G core network [11]. However, using passive measure-

[1] Although other transport protocols exist, the throughput of TCP is the most important practical measure since the vast majority of flows use TCP (e.g., we find that 95% of flows in a large 3G network use TCP).

ments to estimate max-throughput poses unique challenges. First, because the network does not control either end-point, it can not control the duration of TCP flows. TCP flows take several round trips to ramp up to their fair-share capacity, so the observed throughput of small flows may not approximate max-throughput. The duration of TCP slow start is even longer in 3G networks because RTTs are typically 100s of milliseconds [12]. Second, flows may be throttled for reasons other than reaching the available network capacity. For example, content providers may rate limit streaming content. Finally, the enormous traffic volume of all TCP flows passing through a large network means that it is impractical to record and analyze every packet. In practice, only periodic flow samples can be recorded, and little processing can be done on those samples if we wish to monitor max-throughput in near real-time.

In this paper, we develop a novel technique to estimate max-throughput using only passively measured flow records. Our technique works by constructing a *Throughput Index* (TI) that includes only flow types that accurately estimate max-throughput. Only flows that match a flow type in the TI are used in estimates. We validate our technique in a large 3G wireless network by comparing our passive max-throughput estimates with active measurements. We make the following contributions:

- Compared to previous passive TCP parameter estimation techniques [2, 10, 11, 13, 14], our approach does not require packet traces and requires no online processing of flow records except for a simple, constant time filter. Therefore, our approach is amenable to very large-scale max-throughput monitoring and is already deployed as a prototype to monitor all data traffic in a large 3G network.

- To our knowledge, our approach is the first passive technique to estimate max-throughput that is validated on traffic in the wild. We find that our max-throughput estimates correlate well with measurements from active probes in several large metropolitan areas. In aggregate, we find a correlation coefficient of 0.88.

- Through the construction of a TI, we classify rate-limited content-providers and applications. Surprisingly, we find that nearly 60% of large flows are rate-limited and can never reach the peak network capacity.

2. BACKGROUND

To evaluate passive max-throughput estimation, we analyze traffic traces from a large UMTS wireless network. The majority of our analysis in this paper uses traces collected from April 1-7, 2010. We collected these traffic traces from an infrastructure [4] that monitored all traffic on the Gn interface between all Serving GPRS Support Nodes (SGSNs) and Gateway GPRS Support Nodes (GGSNs) in the packet-switched part of the UMTS Core Network [7]. Data traffic that originates from user handsets travels from the Radio Access Network (RAN) to the Gn interface before exiting to the Internet. Similarly, all traffic from the Internet to user handsets crosses the Gn interface. These traces contain traffic from all regions of the UMTS network roughly within the Pacific and Central timezones. The data that we use for this study does not contain personally identifiable information.

Flow Records. For our study, the measurement infrastructure collected one *flow record* for each flow every 1 minute for a random 3% of users. Flows are distinguished by the standard $(ipsrc, ipdst, sport, dport)$ tuple. For the purposes of our study, each flow record is annotated with two fields: *application*, the application protocol used in the flow, and *content-provider*, the service the flow is communicating with. Each record is also annotated with the following three statistics: *bytes*, the volume transferred during the 1 minute interval, *duration*, the time between the first and last packets in the interval, *total_bytes*, the volume transferred since the start of the flow. Note that the annotated flow records contain no personally identifying information. The *application* classification in our annotated flow records uses well known heuristics based on application headers and port numbers (see [6] for details). The *content-provider* for the annotation is identified by the HTTP Content-Provider header or the DNS name of the server when available. It is empty otherwise. HTTP traffic dominates the traffic as it is used for web browsing, downloads, and streaming. We define a *flow type* to be a (*application*, *content-provider*) pair; we find it useful to group flows records by flow type.

Device Categories. Different handset types have maximum air interface speeds that differ by several Mbps. In order to factor out the influence of different handset types and radio access technologies, this paper only considers downlink 3G flows from HSDPA category 6 devices, which are able to reach 3.6 Mbps in the download direction. In practice, we use our technique to estimate max-throughput for each class of devices and in each direction separately. The device type is identified by the Type Allocation Code of the device, which is available in the GTP tunnel carrying IP traffic between the GGSN the RAN [1].

Throughput Normalization. For proprietary reasons, all throughput values presented in this paper are normalized by an arbitrary constant (i.e., $normalized\_throughput = throughput/C$). Normalization does not change the dynamic range represented in figures.

3. METHODOLOGY

Our goal is as follows: given all TCP flow records that traversed a set of network paths during a time interval, output an estimate of the average max-throughput over that time interval when downloading from an unconstrained Internet source. The most trivial algorithm would be to apply a summary function over the $byte/duration$ values in all flow records (e.g., the mean). However, most of these values will not reflect max-throughput because bytes transferred depends not only on available capacity, but also on total flow size, application protocol, and content-provider. This section describes our technique for filtering flow records to discard the effects of these other factors.

3.1 Why not Measure All Large Flows?

A TCP flow can transfer many bytes before achieving its steady-state throughput because it begins in a *slow-start* phase that incrementally probes for the available capacity. Therefore, an obvious first step to measure max-throughput is to only examine flows that have transferred enough bytes to exit slow-start. The volume transferred during slow-start depends on the RTT and the bottleneck capacity. These two parameters vary based on flow, but a first order ap-

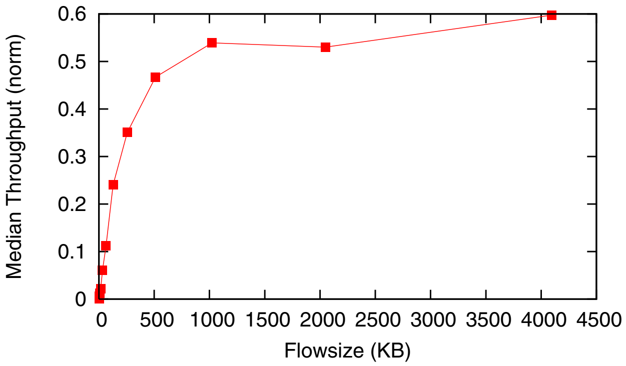

Figure 1: Median normalized throughput of non-rate-limited flow records vs. flow size. All flow records with size $2^i \leq total\_bytes < 2^{i+1}$ are aggregated in the bin 2^i.

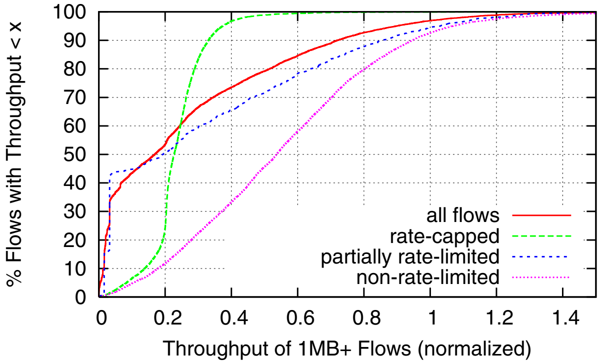

Figure 2: CDF of the measured throughput distribution of 1MB+ flows from: all flows, a (*application*, *content-provider*) that is rate-capped, one that is partially rate-limited, and one that is not rate-capped or partially rate-limited.

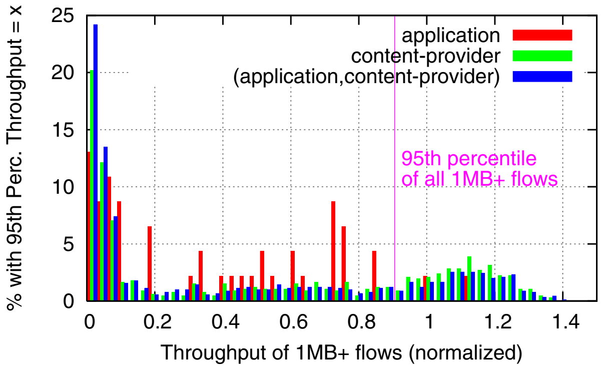

Figure 3: Histogram of the 95th percentile of throughput from each *application*, *content-provider*, and (*application*, *content-provider*) flow type, respectively.

proximation is to find a flow size that allows most flows to exit slow-start.[2] Figure 1 shows that the median measured throughput of non-rate-limited flow records stabilizes at about flow size = 1MB. (We describe how we identify rate-limited flows in §3.2.)

Thus, the next most obvious algorithm is to apply a summary function over the *byte/duration* values in all flow records that have $total\_bytes \geq$ 1MB. For brevity, we call these records the *1MB+ flows*. However, this algorithm is not sufficient because measured throughput of identically sized large flows can still vary based on application protocol and content-provider. For example, Figure 2 shows the distribution of measured throughput values over 1MB+ flows for several (*application*, *content-provider*) flow types. The rate-capped type appears to be bottle-necked at or rate-limited by the content-provider since none of its flows achieve the higher throughputs possible, as shown in the tail of the all flows line. This is in contrast to the non-rate-limited type, which achieves throughput values throughout the possible spectrum. The partially rate-limited line shows that a flow type can exhibit bimodal behavior: rate-limited in some flows (0-40%) but non-rate-limited in others (40-100%). This can be because the same application protocol is used for control messages and bulk transfer. Our inspection of flow types showed that most fall into one of these three categories. This includes application protocols that are used only for control messages, which would appear to be rate-capped at a very low throughput value.

Note that a flow can appear to be rate-limited for a variety of reasons, including traffic shaping by the content-provider, application protocol bottlenecks, and persistent congestion or capacity problems on the Internet path to the server. In practice, we do not actually need to detect the cause of the rate-limiting, only its effect on the throughput distribution.

3.2 Identifying Rate-Limited Flows

To obtain a more accurate measure of max-throughput, we must filter out the applications and content-providers that have flow distributions similar to the rate-capped and partially rate-limited flow types. We define a *rate-capped* flow type to be one that never reaches the available capacity of the network. We defined a *partially rate-limited* flow type to be one that has a significant fraction of rate-limited flows. We describe two heuristics to detect each flow type below.

To identify rate-capped flows, we note that the rate-capped flow distribution shown in Figure 2 never crosses the tail of the all flows distribution. In general, if we assume that at least 5% of all 1MB+ flows reach the available capacity, then a non-rate-capped flow type should have a 95th percentile throughput at least as large as the 95th percentile throughput of all 1MB+ flows. This is because all 1MB+ flows includes both rate-limited and non-rate-limited flow records. Figure 3 shows a histogram of the 95th percentile of each flow type, where we define flow type by *application* only, *content-provider* only, and (*application*, *content-provider*) pair. Only flow types with at least 100 flows are presented.

[2] Since our study examines a wireless network, TCP may exit slow start due to physical layer loss rather than congestion induced loss. Indeed, the throughput in Figure 1 continues to increase linearly at a slow rate even after the knee in the curve, which indicates that TCP AIMD sometimes finds more capacity available after slow-start. However, since our goal is to estimate the max-throughput that real users actually experience, we do not attempt to exclude the effects of physical layer loss.

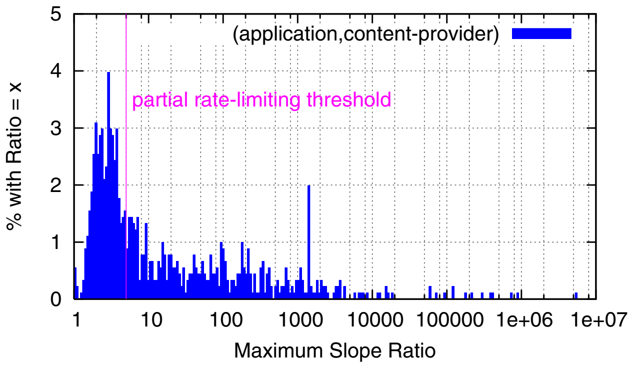

Figure 4: Histogram of the Maximum Slope Ratio of each (*application*, *content-provider*) flow type for details. See text in §3.2. Note the logarithmic scale of the x-axis.

We see that for the *content-provider* and (*application*, *content-provider*) distributions, there is a clear mode to the right of the "95th percentile of all 1MB+ flows" line. This mode represents the non-rate-capped flows. The *application* distribution does not exhibit this mode, which suggests rate-capping is primarily a property of content-providers, not application protocols. We identify all flow types to the left of the line as rate-capped.

To identify partially rate-limited flows, we note that the partially rate-limited flow distribution shown in Figure 2 is bimodal: the first portion of the distribution has a very steep slope due to rate-limiting, while the later portion is less steep. In general, we want to detect these dramatic decreases in a flow type's CDF slope. We use the following heuristic: Let s_i and s_{i+5} be the slopes at percentile i and $i + 5$, respectively. We define the *slope ratio* of s_i and s_{i+5} to be s_i/s_{i+5}. Define the maximum slope ratio to be the greatest slope ratio over $i \in [7, 8, 9, \ldots, 93]$ (we ignore the top and bottom percentiles to guard against outliers). The maximum slope ratio will be large if there is a dramatic decrease in slope within any 5 percentile range. In practice, we approximate s_i as the difference between percentile $(i - 2.5)$ and percentile $(i + 2.5)$.

Figure 4 shows a histogram of the maximum slope ratio over each (*application*, *content-provider*) flow type. Again, only flow types with at least 100 flow records are presented. We see a primary mode to the left of line at maximum slope ratio = 5. This mode represents flow types without dramatic changes in slope. However, there is also a long tail to the right of this line. We identify flow types to the left of the "partial rate-limiting threshold" as partially rate-limited. The choice of a threshold = 5 conservatively captures the most of the flow types in the main mode. We note that our approach described in the next section is not very sensitive to this threshold value (we have tried thresholds up to 20 without noticeable differences).

3.3 Our Approach: A Throughput Index

To obtain a better estimate of max-throughput, our approach is to include only flow types that are non-rate-capped and non-rate-limited. We call the set of all (*application*, *content-provider*) types that satisfy these criteria the *Throughput Index* (TI). Table 1 shows the top 15 flow types by number of 1 MB+ flows, whether they are identified as rate-capped (C) and/or partially rate-limited (L), and their mean and median throughputs. The bold entries are included in the TI. We see that mean and median throughputs of TI flow types are much closer to each other than non-TI flow types, as expected of unconstrained vs. constrained downloads. One anomaly is the "web browsing" flow type third from last in the table, which has mean and median throughputs similar to the other TI flow types. A few of these potential "false negatives" exist because their 95th percentiles of throughput or maximum slope ratios fall just below or above our choice of thresholds. However, it is clear from Figure 3 and Figure 4 that modifying the thresholds slightly would not dramatically change the fraction of rate-capped or partially rate-limited flow types. Moreover, a perfect classification isn't necessary for the TI to be functional.

			% 1MB+	Throughput (norm)	
Flow type	C	L	Flows	Median	Mean
(streaming, C1)	●		11.79	0.229	0.237
(http other, C2)			**10.91**	**0.497**	**0.502**
(file download, C3)			**9.35**	**0.427**	**0.477**
(http other, C4)	●	●	6.78	0.032	0.028
(streaming, C5)			**6.36**	**0.539**	**0.559**
(p2p, C6)	●	●	3.03	0.014	0.029
(unknown, C7)	●	●	2.80	0.009	0.027
(email, C8)	●		1.19	0.056	0.116
(web browsing, C4)	●	●	0.95	0.032	0.026
(streaming, C5)		●	0.92	0.266	0.328
(http other, C9)			**0.90**	**0.566**	**0.576**
(email, C10)	●		0.80	0.147	0.175
(web browsing, C2)	●		0.76	0.421	0.415
(streaming, C4)	●	●	0.75	0.025	0.028
(email, C11)	●		0.61	0.072	0.155
...					
TI			38.7	0.503	0.524

Table 1: Top 15 (*application*, *content-provider*) flow types, the % of 1MB+ flows they comprise, and the normalized median and mean throughputs of their 1MB+ flows. Each content-provider is identified with a consistent C*x* identifier. Column 'C' and 'L' indicate rate-capped flows and partially rate-limited flows, respectively. Bold rows are in the TI.

Table 1 also shows that inspecting the application protocol or content provider is not sufficient to determine which flow types are rate-limited. For example, the top two streaming applications use the exact same streaming protocol, but one is clearly rate-capped while the other is not. This is because the protocol can be configured so that the entire stream is downloaded at once. In addition, we see that identifying flow type by (*application*, *content-provider*) rather than just *application* or *content-provider* is important, since some content-providers have both non-rate-limited and rate-limited applications (e.g., C2 and C5). Table 2 shows the percentage of flows and flow types in each flow type category. Only the flow types with at least 100 flows are analyzed, but the remaining flow types comprise only 6% of flows. Surprisingly, nearly 60% of large flows are rate-capped and can never reach the peak network capacity. The TI includes 39% of flows and 23% of flow types.

In practice, we compute the TI offline based on a representative time period of flow records. Once computed, we process flows online using the TI as a filter to select flows for max-throughput estimates. We currently recompute the TI

	C	L	C+L	TI
% 1 MB+ Flows	25.8	1.9	33.5	38.7
% (*application*, *content-provider*)	24.3	3.0	49.6	23.1

Table 2: Percent of flows and (*application*, *content-provider*) types that are rate-capped (C), partially rate-limited (L), both (C+L), and in the TI.

once every few months since we have observed that the distributions of popular (*application*, *content-provider*) types do not change often.

A final question is how to aggregate the *byte*/*duration* measurements of flows in the TI. We evaluate two approaches: The first approach, TI-F, takes a mean over the throughputs of all flows records in the TI. This aggregate will be relatively robust to outlier users since it weights a very large number of flows from different users equally. However, it is also sensitive to non-network problems that impact the top 3 content-providers since they make up a majority of all flows in the TI. The second approach,TI-T, takes the mean of the means of each flow type. This aggregate weights each flow type equally so it is more robust to unexpected changes with individual content-providers, but it is more sensitive to unpopular flow types that may only be used by a small number of users. In the next section, we show that the estimate produced by each of these aggregates is comparable under typical circumstances.

4. EVALUATION

Evaluating the accuracy of any max-throughput estimation technique, whether passive or active, is difficult because "ground truth" measurements are not available from all user locations at all times. In this section, we evaluate our passive max-throughput estimation techniques by comparison with a set of active measurements. Although these active measurements do not necessarily represent ground truth, they do represent the current state-of-the-art for max-throughput estimation. The TI estimates of max-throughput are closer to these active measurements than alternative passive measurement techniques, which suggests that the TI estimates are more representative of max-throughput.

4.1 Setup

In this section we compare our passive max-throughput estimation techniques against active measurements in several 3G wireless network regions. Each region roughly covers a major metropolitan area. In addition, we compare against the aggregated measurements from all regions in roughly the Pacific and Central timezones.

We perform active throughput measurements from probes in several stationary locations. Each probe performs a throughput measurement by downloading a 3MB file via FTP from a well-provisioned server close to the Gn interface. 2 to 3 measurements per probe are collected each hour. Each region we consider in this section has probes in 3 to 12 different vantage points. The active max-throughput estimate we report each hour is the mean of all measurements from all probes in a region. We note that the active probes are generally placed in locations with good RF conditions. Thus, we expect that they would perform better than the average subscriber handset.

We select the flow types in the TI using traffic from March 24-30, 2010. Then, we compute the TI-F and TI-T estimates based on traffic observed on April 3-7, 2010. We also compute the naïve All 1MB+ Flows estimate, which corresponds to the mean throughput of all flow records with *total_bytes* $\geq$ 1 MB. We compare these passive max-throughput estimates to the active estimates for each hour during the same time period.

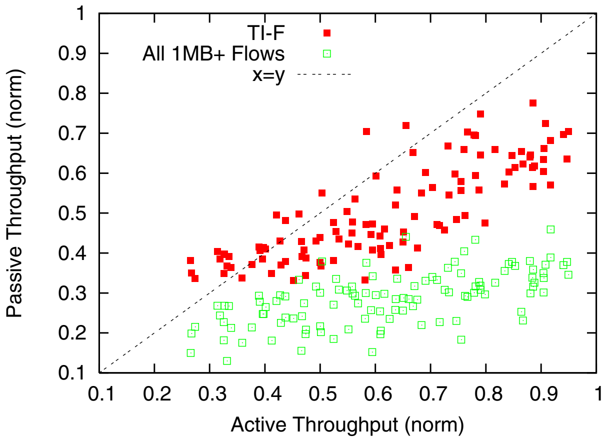

Figure 5: Comparison of active and passive estimates for region R1. Each point represents the estimate for one hour.

4.2 Results

Figure 5 shows a scatter plot comparing active and passive estimates. Each point represents the estimate for one hour in the largest region. If active and passive estimates report the same values, then the points would fall on the x=y line. We see that the All 1MB+ Flows approach produces estimates that are significantly less than the active measurements. TI-F produces estimates that are much closer, but are still generally less. This may be because some flows in the TI are still rate-limited by application behaviors that we do not detect. It may also be because the active measurement probes are in higher quality vantage points (i.e., better RF conditions) than most real users. During a few hours, the TI-F estimate is higher than the active measurement (i.e., the points above the diagonal). These cases can probably be attributed to variance in the small number of active measurement samples. Evaluating which set of measurements is closer to the "ground truth" is the subject of future work, but we are encouraged that both active estimates and the TI-F estimates show a similar trend over time.

Relative Difference. To see if this trend generalizes, we compare the relative difference between the passive and active estimates in other regions. Figure 6 compares the relative difference between each set of passive and active estimates for all regions and the 10 regions with the most active probe vantage points. The top of each bar indicates the median relative difference (over all hours) and the error bars show the 25th and 75th percentiles. We see that both the TI-F and TI-T estimates have roughly the same relative difference over all regions and both have relative differences substantially less than the All 1MB+ Flows approach. Most TI-F and TI-T estimates are less than 30% different than the active measurements, while most All 1MB+ Flows estimates

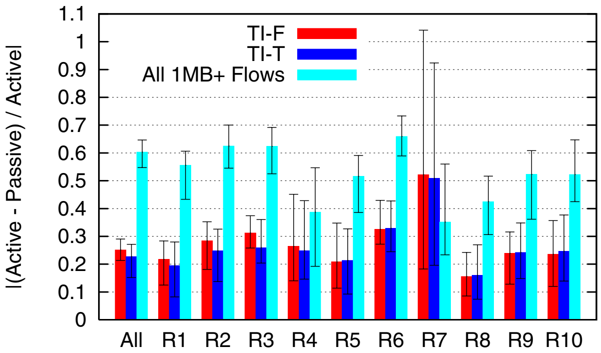

Figure 6: Relative difference of passive estimates to active throughput estimates in all regions and the 10 regions with the most active probe vantage points. That is, $|(active - passive)/active|$. The top of each bar shows the median relative difference over all hours and the error bars show the 25th and 75th percentiles.

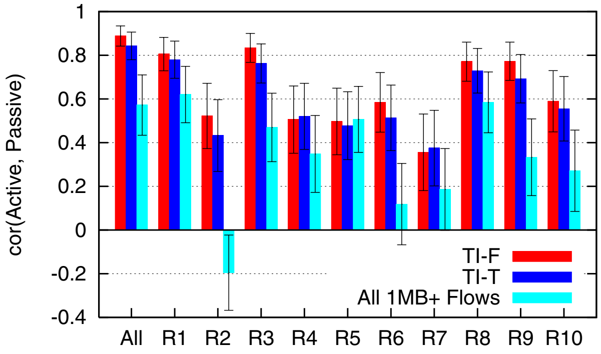

Figure 7: Correlation of passive estimates to active throughput estimates in all regions and the top 10 regions. Each error bar shows the 95% confidence interval of the corresponding correlation coefficient.

are more than 50% different. One anomaly is region R7, where the TI estimates have greater relative differences. We believe this is due to the active probes being in unrepresentative locations because this is the only location where the active estimates are lower than all passive estimates, on average.

Correlation. In addition to similar estimate values, we also expect good passive estimates to be correlated with the active estimates over time. That is, when the active estimate goes down (e.g., due to contention) we also generally expect the passive estimate to go down. Figure 7 shows Pearson's correlation coefficient between each passive estimate time series and the corresponding active estimate time series in all regions and the top 10. The error bars show 95% confidence intervals of the correlation coefficients. Two perfectly correlated signals would have a correlation of 1 and any correlation greater than 0.6 is well correlated. We see that both TI-F and TI-T are at least as correlated with the active estimates as the All 1MB+ Flows estimates. The correlation is substantially greater than the All 1MB+ Flows estimates in some regions, such as R2 and R6.

The regions where the TI estimates are less correlated with the active estimates, such as R2, R4, R5, and R7, are regions where there are fewer samples either in the TI or from the active probes. When many vantage points are aggregated, such as in the All case, the TI and active estimates' correlations are very high (close to 0.9). This suggests that when enough samples are available, the TI estimates correlate well to aggregate network-level effects such as shared congestion. To improve the TI's correlation of these effects at finer network granularities, we are currently increasing the sampling rate of flow types in the TI from 3% to 100% of users. This is feasible because the number of TI flows is small relative to the total flows that traverse the network.

5. CONCLUSION AND FUTURE WORK

Our results demonstrate that max-throughput can be estimated using passive measurements via judicious selection of flows. In this paper, we presented our initial attempt at such a selection by identifying non-rate-limited flow types to place in a Throughput Index.

By applying the TI approach to more real traffic, we hope to resolve a few outstanding issues. First, the the minimum flow size necessary to reach TCP steady-state depends on the RTT and the available capacity, both of which are dynamic quantities. We plan to explore how this flow size can be varied based on network conditions, which should improve the TI's max-throughput estimates when available capacity grows. Second, the filtering of certain flow records presents a trade-off between the number of samples and their aggregate accuracy. We plan to explore how to utilize the noisier rate-limited samples when an insufficient number of non-rate-limited samples exist. Third, application protocol and content-provider behavior can change over time. We plan to explore how to detect such changes dynamically by examining how each flow type contributes to the TI over time. Fourth, our approach assumes that most flows in the TI are typical and benign. In the future, we will explore how malicious and abnormal flows that skew the max-throughput estimate can be detected. Finally, while we believe the TI approach generalizes to wired networks, further study is needed to understand the impact of the greater heterogeneity in vantage points and TCP stacks.

6. REFERENCES

[1] 3GPP. Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); General Packet Radio Service (GPRS); GPRS Tunnelling Protocol (GTP) across the Gn and Gp interface (3GPP TS 29.060 version 6.18.0 Release 6). ETSI TS 129 060 V6.18.0 (2007-10), 2007.

[2] J. But, U. Keller, and G. Armitage. Passive TCP Stream Estimation of RTT and Jitter Parameters. In *LCN '05: Proceedings of the The IEEE Conference on Local Computer Networks 30th Anniversary*, pages 433–441, Washington, DC, USA, 2005. IEEE Computer Society.

[3] R. L. Carter and M. E. Crovella. Measuring bottleneck link speed in packet-switched networks. *Perform. Eval.*, 27-28:297–318, 1996.

[4] C. Cranor, T. Johnson, O. Spatscheck, and V. Shkapenyuk. Gigascope: A stream database for network applications. In *SIGMOD '03: Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, pages 647–651, New York, NY, USA, 2003. ACM.

[5] C. Dovrolis, P. Ramanathan, and D. Moore. Packet-dispersion techniques and a capacity-estimation methodology. *IEEE/ACM Trans. Netw.*, 12(6):963–977, 2004.

[6] J. Erman, A. Gerber, M. T. Hajiaghayi, D. Pei, and O. Spatscheck. Network-aware forward caching. In *WWW '09: Proceedings of the 18th international conference on World wide web*, pages 291–300, New York, NY, USA, 2009. ACM.

[7] H. Kaaranen, S. Naghian, L. Laitinen, A. Ahtiainen, and V. Niemi. *UMTS Networks: Architecture, Mobility and Services*. Wiley, New York, NY, 2001.

[8] R. Kapoor, L.-J. Chen, L. Lao, M. Gerla, and M. Y. Sanadidi. CapProbe: A simple and accurate capacity estimation technique. *SIGCOMM Comput. Commun. Rev.*, 34(4):67–78, 2004.

[9] S. Katti, D. Katabi, C. Blake, E. Kohler, and J. Strauss. MultiQ: automated detection of multiple bottleneck capacities along a path. In *IMC '04: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, pages 245–250, New York, NY, USA, 2004. ACM.

[10] J. Pahdye and S. Floyd. On inferring TCP behavior. In *SIGCOMM '01: Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 287–298, New York, NY, USA, 2001. ACM.

[11] F. Ricciato, F. Vacirca, and M. Karner. Bottleneck detection in UMTS via TCP passive monitoring: a real case. In *CoNEXT '05: Proceedings of the 2005 ACM conference on Emerging network experiment and technology*, pages 211–219, New York, NY, USA, 2005. ACM.

[12] P. Romirer-Maierhofer, F. Ricciato, A. D'Alconzo, R. Franzan, and W. Karner. Network-Wide Measurements of TCP RTT in 3G. In *TMA '09: Proceedings of the First International Workshop on Traffic Monitoring and Analysis*, pages 17–25, Berlin, Heidelberg, 2009. Springer-Verlag.

[13] S. Seshan, M. Stemm, and R. H. Katz. SPAND: Shared Passive Network Performance Discovery. In *USITS'97: Proceedings of the USENIX Symposium on Internet Technologies and Systems on USENIX Symposium on Internet Technologies and Systems*, pages 13–13, Berkeley, CA, USA, 1997. USENIX Association.

[14] M. Zangrilli and B. B. Lowekamp. Applying Principles of Active Available Bandwidth Algorithms to Passive TCP Traces. In *Passive and Active Network Measurement*, pages 333–336, 2005.