Combined Terminal and Network Measurement System for Bottleneck Localization

Pekka H. J. Perälä, Marko Jurvansuu, and Jarmo Prokkola

Converging Networks Laboratory

VTT Technical Research Centre of Finland,

Kaitoväylä 1, FIN-90571 Oulu, Finland

firstname.lastname@vtt.fi

ABSTRACT

When a user experiences a poor quality of service (QoS) in a converged network environment, the operators or ISPs need to locate the bottleneck, i.e. to localize which part of the end-to-end (E2E) network path causes the degradation. We present a novel multipoint measurement system that gathers performance information not only from links in the network path, but also from the end devices. In wireless domain, this enables to separate the radio access network and fixed network segments. The measurement method is also applicable to pure fixed line networks. The system is able to perform one-way measurements and even break the E2E delay into one-way link-to-link components, which gives clear benefit over traditional round-trip time (RTT) measurements. The system was verified with laboratory 3G/HSPA (High Speed Packet Access) measurements that revealed a performance bottleneck in the studied 3G network. We also show measurement results in a live Wi-Fi city area network.

Categories and Subject Descriptors

C.2.3 [**Network Operation**]: *network management, network monitoring, public network.*

General Terms

Measurement, Performance.

Keywords

QoSMeT; end-to-end; HSPA; 3G; WLAN; GPS.

1. INTRODUCTION

The usage of multimedia applications over high speed data networks has become an important part of consumers' everyday life. Traditionally homes and offices have broadband connections with last mile fixed line technologies such as fiber or Digital Subscriber Line (xDSL). Nonetheless, the recent advances in

wireless technologies have created possibilities to provide broadband to consumers also with Wi-Fi, WiMAX, or even cellular technologies via HSPA enhancement among others.

However, a broadband connection does not solve all the needs that a user may have. As a matter of fact, the perceived quality is sometimes dissatisfactory, especially with users of real-time applications such as online games, although the nominal bit rate of the connection would be sufficient. In addition, the performance of fixed xDSL line to homes is not always sufficient for bandwidth demanding web-casts or IPTV-traffic. In the wireless domain, the issue of providing large bandwidth with low delay and jitter is even more challenging.

At present, operators, ISPs, or service providers face a challenge to respond to user complaints regarding the end-to-end connection quality. In many cases, there is no clear view whether the problem was originated from the part in end-to-end path that they possess or is of someone else's. Usually one company does not have administrative access to all nodes in the path. The path may also include both fixed and wireless links, which have different kinds of monitoring tools and interfaces. It is also difficult to determine from the network side the level of quality degradation perceived by the user. Among many things, this depends on the user equipment and the application specific needs for communication quality. For example, voice quality for voice over IP (VoIP) is sensitive to packet loss and delay, whereas for video, available bandwidth is important as well. One also needs to address the possible performance asymmetry in the uplink (UL) and downlink (DL) directions, which makes the perceived quality different at the end-points.

Our research aim has been to develop a solution that aid operators and service providers to clarify challenges of the E2E path. This requires identifying the problematic link in real-time using multiple measurement points in the path. Also the user application traffic flows need to be tracked in both directions. Thus, our aims have been to provide a system capable of a) passive measurements with traffic filtering, i.e. any application can be measured, b) multipoint measurements that enables locating bottleneck links, c) composition of terminal and network measurements in order to have both viewpoints, d) accurate one-way delay measurements to deal with asymmetrical links, and e) interoperability with commercial tools.

There are a number of methods and tools presented in the literature for bandwidth bottleneck localization [1-5]. However,

they mainly focus on the overall performance of the network, whereas we try to provide accurate one-way QoS measurements and e.g., delay bottleneck localization for individual user related traffic flows. Furthermore, the existing solutions use often active measurement methodology, whereas we consider passive measurement approach advantageous. From our point of view, the performance of a single application is very interesting, since it represents the user perceived quality.

This paper presents a novel system for delay bottleneck localization by using a multipoint measurement technique. The measurement system is not tied to any particular network infrastructure, but is applicable to almost any network using IP. At the moment, we have performed measurements over Ethernet, ATM, Wi-Fi, WiMAX, and cellular technologies. The multipoint measurement is applied to the E2E path by calculating QoS values based on traces taken in end-points and also in one or more strategic intermediate points. Global Positioning System (GPS) assisted timestamping in each measurement point (MP) enables accurate delay measurements. In addition to measuring application specific traffic flows of a single user, the system is able to make assessment of application dependent quality. At the moment, VoIP subjective quality estimation is supported. Another feature is the possibility to share measurement information between the measurement points. This way, the person conducting the measurement can monitor simultaneously not only the end-to-end performance, but also the network segment related values.

The rest of the paper is organized as follows: In Section II we present the multipoint measurement system architecture. In Section III, we show measurement results and in Section IV we conclude the work and discuss on future work.

2. MULTIPOINT MEASUREMENT SYSTEM

As the name 'multipoint measurement system' implies, the MPs are set up in multiple locations. Figure 1 shows the overall principle how measurement points are located on the IP traffic source, intermediate points in network path, and on the IP traffic destination. The end-points can be e.g., terminals, services, or network devices. Boundaries or gateways between the network segments are usually favorable places to monitor, since all the traffic flows through them.
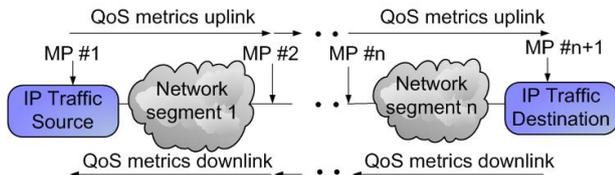

Figure 1. Multipoint QoS measurement system principle.

The multipoint measurement point system is a result of evolution from our E2E measurement tool towards a combined terminal (end-to-end) and network (intermediate nodes) measurement system. The heart of the system is software called Multipoint QoS measurement data Collector (MQC). It controls the MPs, captures desired traffic, and calculates performance metrics. It also does some pre-analysis to the results in real-time. The

communication between the measurement points is carried out by a special control connection. The amount of control traffic increases as a function of the number of MPs. Thus, control traffic load was designed to be as low as possible. If very low-bandwidth networks are monitored, the control traffic between the MPs can be routed via a path other than the inspected one. Another option is to store the measurement data locally and distribute it after the measurement. The system runs on most Win32 based systems.

In the terminal side we use an end-to-end QoS measurement tool called QoSMeT developed earlier by VTT [8]. It is a real-time passive measurement tool that captures data link level packets and uses GPS-based timestamping for accurate delay measurements. The typical way to use QoSMeT is that the traffic of some interesting application(s) is tracked giving direct information on how the application sees the network performance. Measurements at intermediate points are performed with either QoSMeT clients or the commercial M5 protocol analyzer by NetHawk [9]. The M5 analyzer is used to capture traffic from cellular network interfaces such as Iub (ATM), but it can also be used to measure Ethernet/IP based links.

All measurement points can be synchronized with GPS to allow accurate delay measurements between MPs. Other measurable quantities are e.g., throughput, offered load, jitter, packet loss, connection break length, and traffic statistics. The measurement accuracy depends on the used timestamping method as summarized in Table 1. The GPS was chosen as the reference time source, because of its clearly better accuracy when compared e.g. to NTP [10]. In E2E measurements, the instantaneous accuracy is limited by VTT's QoSMeT GPS driver accuracy, which is better than 44 μs. The manufacturer gives accuracies better than 1 ms and 10 μs for ATM and Ethernet adapters, respectively. This means that if the ATM adapter is used at the intermediate measurement points, it limits the total accuracy. The accuracy of 1 ms is sufficient to measure networks whose delays are in order of tens of milliseconds, but for high speed networks (Ethernet etc.) it is not enough. QoSMeT GPS driver accuracy, however, is sufficient for our purposes to measure also high speed networks. More information on the accuracy of our system can be found in [11].

Table 1. Timestamping accuracies

Timestamping method	Absolute accuracy
QoSMeT GPS driver	< 44 μs
ATM adapter	< 1 ms
Ethernet adapter	< 10 μs

By measuring QoS metrics, on both end and intermediate nodes, it is possible to locate performance bottlenecks on the E2E path. There are several solutions (see [1-5] and references therein) for localization of bandwidth bottlenecks, but many real-time applications are more sensitive to other QoS metrics, such as delay and packet loss. Therefore, the link that constitutes most of the E2E delay should be considered as delay bottleneck, which may be a different link than the one with lowest available bandwidth i.e. bandwidth bottleneck. In addition to delay bottlenecks, also the bandwidth bottlenecks can be found by using self-induced congestion principle [5]. According to the

principle, the bandwidth bottlenecks can be found by simply increasing the offered traffic load until the maximum bandwidth B_{max} of the system is reached. When the maximum is exceeded, the bottleneck node starts to queue packets, which leads to increased delay and finally to increased packet loss as well. In this way it is possible to measure the maximum bandwidth of the E2E path at the same time the delay bottleneck is localized. However, it is good to keep in mind that often, in addition to the localization, in-depth analysis is needed for finding the reason why a particular link is the bottleneck.

The calculation of QoS values is done in the QoSMeT clients based on the information provided by the end points and intermediate nodes. The delay calculation is simply done by subtracting destination and source timestamps from each other on a per packet basis, while packet loss is determined as

$$P_{loss} = (1 - N_d/N_s) \cdot 100\%, \qquad (1)$$

where N_d denotes the amount of packets received at destination and N_s the amount sent from the source, respectively.

3. MEASUREMENT RESULTS

We present two sets of measurements; Section A describes the cellular network scenario and Section B the Wi-Fi network scenario.

3.1 Bottleneck identification in laboratory 3G network

The first time the system was used for solving a real-life performance problem was with VTT's Converging Networks Laboratory's [12] laboratory 3G/HSDPA network. For some time, HSDPA had suffered from low throughput and occasional instability under high traffic loads. The maximum throughput was roughly 800 kbps, which is quite far from the theoretical 1.8 Mbps for HSDPA equipment based on category 12. Hence, it provided a good testing platform for the new multipoint measurement system. The laboratory 3G network consists of a real nodeB that is connected to RNC/Iub simulator, which in turn is connected to VTT's laboratory LAN. Commercial terminals that have VTT's SIM cards can attach to the network and start either circuit switched calls inside the RNC simulator area or packet data sessions to VTT's laboratory LAN or to Internet.
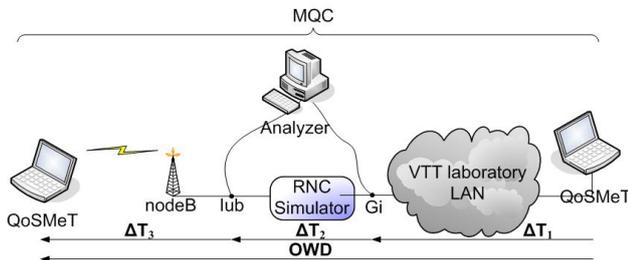

Figure 2. Measurement setup for delay bottleneck localization in VTT's laboratory 3G network.

The measurement setup is illustrated in Figure 2. One end point was attached to a laptop with HSDPA data card and the other one was attached to VTT's laboratory LAN. The intermediate measurement points were attached to Iub interface capturing ATM/IP traffic and to Gi interface, which captured Ethernet/IP

traffic. The intermediate points were chosen in such a way that the E2E performance could be split to LAN (ΔT_1), the RNC simulator (ΔT_2), and the radio access network (ΔT_3) segments. During the measurements we recorded delay, bandwidth, jitter, and packet loss for these network segments. However, due to the limited space, we mainly considered delay in this paper. For the multipoint measurements, it is typical that large amount of data is gathered even from relatively simple measurement scenarios.

Test traffic was generated with D-ITG software [13], because of its good capabilities to produce traffic loads with different distributions. UDP (User Datagram Protocol) traffic was sent from the laptop connected to LAN to DL direction. The destination was a terminal connected to the 3G network using HSDPA. The interarrival times were chosen to follow Poisson process, because a constant sending interval might induce measurement artefacts due to the scheduling period in the Iub flow control [7, 8]. Constant packet size of 1000 B was used. No other users were allowed to access the 3G network to minimize the sources of errors in this test measurement. All the tests were repeated several times with the same settings to increase the statistical reliability.

The goal of the first test sequence was to reveal the steady state performance of the network. The offered load was 320 kbps, which is well below the load that caused network instability based on our experience. The measured average delays and corresponding 95% confidence intervals for each network segment are collected in Table 2. The results show that most of the delay is caused by radio access (28 ms), while LAN segment causes only 3 ms average delay. Both values are typical for these network segments according to our experience. The RNC simulator creates additional 9.6 ms delay for the packets. The E2E packet loss was only 0.08 %.

Table 2. One-way delay statistics for 320 kbps offered load

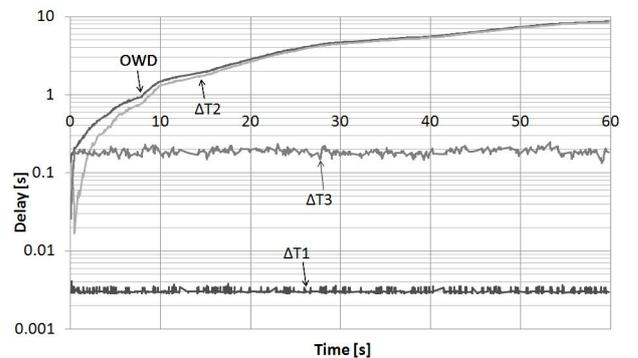|	E2E delay (ms)	ΔT_1 (ms)	ΔT_2 (ms)	ΔT_3 (ms)
Average delay (±95 % confidence interval)	40.69 (±0.18)	3.05 (±0.1)	9.61 (±0.18)	28.04 (±0.18)

Figure 3. Downlink delays with 960 kbps offered load.

In the second test sequence, the offered load was set to 960 kbps. As shown in Figure 3, this time the E2E delay starts steadily to

increase from the start of the measurement and is soon several seconds. This is totally different behavior from the 320 kbps measurement where delay values were roughly the same throughout the measurement duration. The main cause for this is the increase in RNC simulator related delay from steady state 10 ms to delay of several seconds (see also Table 3) Moreover, the radio interface related delay of 185 ms is also clearly higher than in the steady state measurements, but packet loss remained very low also in this case, being about 0.04 %. It is clear that the offered load exceeds the end-to-end maximum throughput, but the results do not give clear answer for the exact bottleneck location. Thus, more measurements were needed.

Table 3. One-way delay statistics for 960 kbps offered load

	$E2E$ delay (ms)	ΔT_1 (ms)	ΔT_2 (ms)	ΔT_3 (ms)
Average delay (± 95 % confidence interval)	4352.29 (±102.83)	3.02 (±0.01)	4163.86 (±102.73)	185.40 (±0.79)

The next measurement sequence was made to pin-point the offered load from which the delays start to increase. This would be the point where the bottleneck would become visible. The measurements were made by offering different loads and measuring average delay over a 60 s time period on each load as shown in Figure 4. The E2E delay started to increase after the offered load was above 640 kbps. As seen, at first, the increase is dominated by radio access part, but RNC simulator delay (ΔT_2) increases quicker and finally, after 800 kbit/s load, RNC becomes the dominating part for high delay, while the radio access segment delay rise settles.
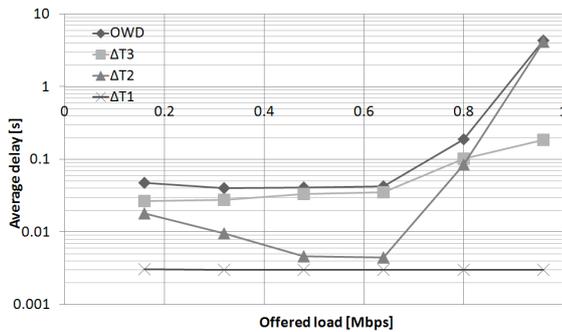

Figure 4. Average delays of network segments as a function of offered traffic load.

As seen also in the figure, slightly surprisingly, RNC segment delay is higher under low traffic load than under moderate traffic load. This interesting behavior encouraged us to perform further analysis of ATM-level packet traces with M5 analyzer. By examining the traces, it was found that Iub flow control behaves in adaptive fashion. In credit based ATM flow control of the Iub, the nodeB tells the simulator (using credits) the amount of data units it can receive in a certain timeframe over the Iub [14]. The amount of credits, which nobeB allows RNC to use in

transmission, is based on the offered traffic load. When examining the case of low traffic of 160 kbps, an interesting observation was made: There was a considerable amount of HS-DSCH (High Speed Downlink Shared Channel) capacity allocation messages. It turned out that due to the bursty nature of the offered traffic, the flow control had trouble in adapting to the average offered load. In particular, it was found that when there were longer interarrival times between generated packets (~ 100 ms), the allowed credits were reset to a small value near zero. Reaching back to adequate level causes extra delay, since the frames have to wait in RNC buffer for the Iub flow control to allow more traffic. On the other hand, with 480 kbps load, the credits were at first increased to sufficient level and remained there. After the start up behavior and during the whole transmission there was continuous adaptation of the credits, but no similar drops in the amount of credits were observed like under low traffic load. With 480 kbps load, the average interarrival times are shorter, and thus, the flow control seems to follow better the average behavior. Thus, the increased average delay under 160 kbps traffic load was caused by the implementation of the Iub flow control.

Anyway, the most important part is considered to be the system behavior under high traffic load. Based on the self-induced congestion principle, our findings would imply that the delay bottleneck would be the RNC simulator segment, since long queuing delays were observed. However, the increased radio access segment delay gives a hint that the cause of the low performance might also be radio related. This can also be seen when carefully observing the first seconds of measurements on Figure 3: The delay of radio interface (ΔT_3) is at first approximately 20 *ms*, but increases rapidly to ca. 200 *ms* in the beginning of the measurement.

The reason for this was likely to be found from RNC/Iub simulator configuration. It contains a large number of parameters that affect the RNC, nodeB, and Iub interface functionalities. Many of these are related to each other and have direct impact on the performance. The detailed analysis of the delay results and Iub interface traces finally revealed that increased delay was caused by the nodeB power settings combined with Iub flow control characteristics. In our case, we finally found out that there was a mixture of settings leading to a too low maximum transmission power of the nodeB. Because of low resources in the radio interface, the nodeB could not deliver all the packets to the terminal in time. This was reflected on the RNC, since the nodeB could not offer enough credits to clear RNC buffers leading RNC to queue the data. Unlike in real RNC, the buffer size in our RNC simulator is unlimited and thus it does not lose any packets. Because of this, more and more packets were situated in the RNC buffer leading queuing time to increase in linear fashion.

The whole bottleneck behavior was finally revealed from the early part delay behavior of Figure 3. In the beginning of the measurement, the nodeB starts to transmit packets to terminal with maximum speed, but the data rate is too low and packets will be queued to the nodeB (ΔT_3 increases). Very soon, the nodeB's buffer gets full (rise in ΔT_3 settles), and nodeB starts to slow down the packet flow by using Iub flow control. As a result, packets will be buffered to RNC (increasing ΔT_2). This

interesting behavior shows that the RNC is the delay bottleneck, although the actual cause for the bottleneck is the radio access part.

After correcting the maximum transmission power for HS-DSCH from 25 dBm to 40 dBm, the previous measurement scenario with different offered loads was repeated. The results shown in Figure 5 reveal that B_{max} is now close to 1.5 Mbps, which is well in line with theoretical 1.8 Mbps [15]. From the figure it can be seen that the queuing behavior is similar to low power measurements, but now shifted to higher data rates. When the maximum bandwidth of the system is exceeded, the simulator starts to queue the packets leading to increased delays.
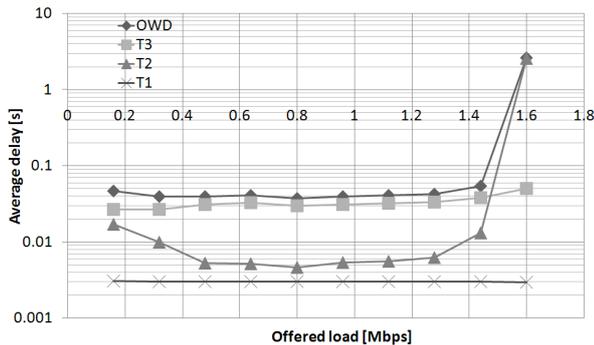

Figure 5. Measured average delays with corrected power configuration.

3.2   Public Wi-Fi network results

In this measurement scenario the multipoint measurement system prototype was applied to public Wi-Fi network called panOulu [16] which has roughly 800 access points (AP) in the city of Oulu, Finland. The goal was to demonstrate the system's measurement capabilities in public network environment and to see how the performance of panOulu is divided between access network and the Internet. For this purpose a measurement setup depicted in Figure 6 was configured. Three measurement points were used allowing us to separate the access network and internet segments. In practice, the Internet path here is the panOulu core network, which is connected to Finnish University IP backbone (FUNET), leading to VTT's laboratory network. The path was verified with the traceroute tool.
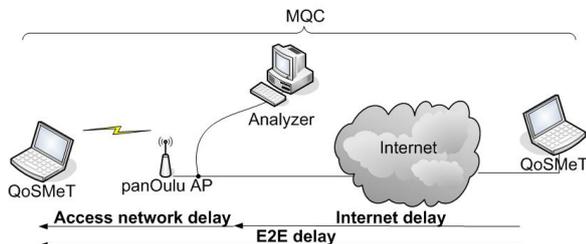

Figure 6. PanOulu measurement setup.

The measurement system was stationary, thus there were no horizontal handovers. The signal quality was on excellent level according to laptop's Wireless LAN (WLAN) client. The advertised nominal maximum data rate was 54 Mbps, using 802.11g. We used AirPcap 802.11 capture device in another laptop to measure the other traffic in the same WLAN channel during the measurement [17]. This was to make sure that there were no other users attached to the same AP during the measurements. We observed that in panOulu, there is continuous broadcast traffic, causing about 10 - 30 kbit/s background traffic also in the AP we were using. For the panOulu core network, we did not have access to make estimates of the instantaneous traffic. The results were verified by conducting similar measurements on different days and on different times of days. The averages were calculated over series of samples whose sizes varied from approximately from 5000 to 65000 samples depending on the offered load.

As with 3G, we conducted first a measurement that gives the steady state performance of the network. We chose VoIP as the measured application traffic, since it is a common service, which is sensitive to delay, jitter, and packet loss. A VoIP call (generating traffic stream of about 18 kbps at application level) was set up with SJPhone application between the two computers. We did not generate any other traffic that would have interfered with VoIP traffic. The results of this measurement are shown in Table 4.

As seen in the table, it is the Internet segment that produces the major component in one-way E2E delay. The wireless LAN performed very well since delays were in the order of a few milliseconds, whereas the Internet part constituted of over 10 ms delay. However, internet delay was quite steady, while the access network's delay caused most of the variation. It was also discovered that the Internet path constitutes most of the E2E packet loss.

Table 4. One-way delay statistics for VoIP call at PanOulu towards DL

	E2E delay (ms)	Access network (ms)	Internet (ms)
Average delay (± 95 % confidence interval)	14.49 (± 0.33)	2.85 (± 0.3)	11.64 (± 0.03)

The next step was to determine how the different offered loads affect the performance and which part will be the bandwidth bottleneck. The different offered loads were induced to network with D-ITG traffic generator similarly as in the 3G network case. The measured traffic was a UDP stream with datagram size of 1 kB. The average delay and E2E packet loss behavior as a function of offered load is shown in Figure 7. In the figure, the secondary y-axis represents the packet loss, while primary y-axis represents delay. As seen, the access network delay starts increasing when the offered load exceeds 9600 kbps. This is accompanied with a rapid increase in packet loss, which is mainly due to the access network segment (radio interface). Packet loss on the internet is constantly clearly less than 1 % (not shown) and delay stays at approximately 12 ms regardless of the traffic load.

The B_{max} that we could reach with one terminal was roughly 13.5 Mbps, which is considerably higher than expected, since according to our network administrators, the link between

panOulu access point and panOulu core network was on purpose limited to 10 Mbps. This was in contradiction with our results for two reasons. First, we observed clearly over 10 Mbps throughput and second, the internet delay did not increase while the offered load was increased over this threshold value. This finding encouraged the administrators to recheck the settings, and indeed, it was found out that there was a 100 Mbps link instead of 10 Mbps, demonstrating the usefulness of multipoint measurements.
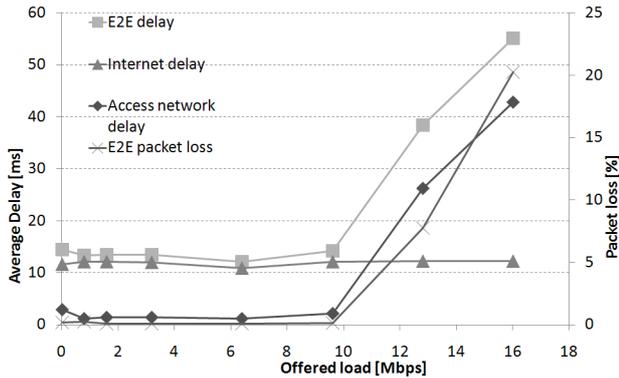

Figure 7. Average delays and E2E packet loss with several offered loads.

However, the observed 13.5 Mbps is clearly lower than the WLAN's nominal data rate of 54 Mbps and also below 25-30 Mbps, which is realistic throughput range to 802.11a (802.11a is very similar 802.11g despite the different operational frequency band) [18]. The peak throughput of 13.5 Mbps was attained with 16 Mbps offered load. However, the average packet loss was already about 20 % with this offered load. With a packet loss of this magnitude many real-time services become unusable [19]. The delay with this offered load for Wi-Fi was also rather high, roughly 40 ms, as can be observed from Figure 8. Also higher offered loads were tried, but it led to increase in packet loss, whilst the throughput did not increase significantly. This kind of behavior is associated to queuing. The buffers are filled when the radio interface cannot deliver all the traffic to the end point. Thus, some percentage of the packets has to be discarded from the buffers. It is likely that the low maximum throughput is at least partially caused by the background traffic. Despite the fact that the background traffic load is quite low, it is composed mainly from very small packets arriving with high rate, and thus causing inefficiency to the random channel access mechanism used by WLAN radio. Another thing is that there is quite much traffic in the neighboring WLAN channels, which was observed with the used spectrum analyzer. Other channels' transmit powers leak to our band causing interference, and possibly causing the Carrier Sense Multiple Access (CSMA) protocol to defer transmissions, finally leading to degraded performance.

Our findings in the panOulu network show that the delay bottleneck resided in the internet part under low offered loads. However, when load was increased, the delay bottleneck changed to the Wi-Fi access segment, and finally, the throughput bottleneck was found to be the WLAN AP. With VoIP traffic and little interference from other users, Wi-Fi delay was in order of milliseconds, which suits well real-time applications' needs.

With offered loads over 10 Mbps, delay and packet loss begun to increase rapidly in the access network segment, as it can be expected, when the maximum capacity is approached, while the internet segment performance remained practically unchanged instead of the varying offered load.
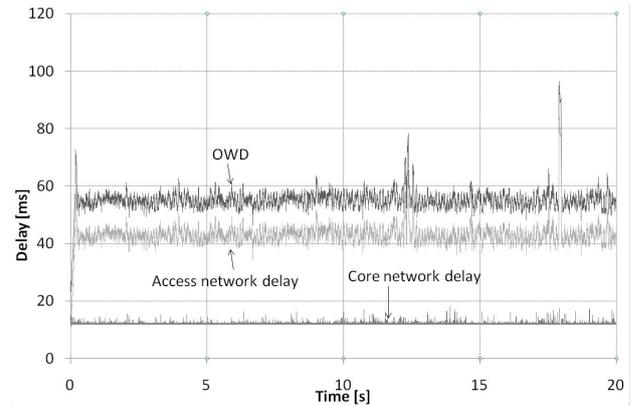

Figure 8. PanOulu delay performance with 16 Mbps offered load.

4. CONCLUSIONS AND FUTURE WORK

In this work we presented our novel multipoint QoS measurement system, which combines terminal and network measurement methods. Although the system is still under development, we were able to show its capabilities to measure passively QoS metrics for a VoIP application. The measurements were done over multiple network interfaces laboratory 3G network and public Wi-Fi network. The results showed how the system can be effectively used in tracing performance bottlenecks. By using the self-induced congestion principle, we were able to locate both delay and bandwidth bottlenecks with some interesting observations.

In the future, the system will be developed further, aiming at more automated testing. A scripting language to create automatic test cases is one option that is considered. Also possibilities for implementing a light-weight QoS module distributed to several locations in the network will be studied.

5. ACKNOWLEDGMENTS

6. REFERENCES

[1] Y. Jiang, C. Tham and C. Ko, "Providing Quality of Service Monitoring: Challenges and Approaches," Network Operations and Management Symposium, April 10-14 2000, pp. 115-128.

[2] V. Visvasuresh, G. Zaruba and G. Balasekaran, "A QoS Scheme To Address Communication Latency Issues For

Critical Network Flows In Best-Effort Networks Using Mobile Agents," Canadian Conference On Electrical And Computer Engineering, May 2-5 2004, pp. 891-896.

[3] M. Palola, M. Jurvansuu and J. Korva, (2004) "Breaking Down the Mobile Service Response Time," 12th IEEE International Conference on Networks, 16-19 November 2004, pp. 31-34.

[4] P. Calyam, D. Krymskiy, M. Sridharan and P. Schopis, "TBI: End-to-End Network Performance Measurement Testbed for Empirical Bottleneck Detection", Tridentcom 2005, February 23-25 2005, pp. 290-298.

[5] V. J. Ribeiro, R. H. Riedi and R. G. Baraniuk, "Locating Available Bandwidth Bottlenecks," IEEE Internet Computing, Vol. 8, Issue 5, pp. 34-41.

[6] F. L. Walls and J.-J. Gagnepain, "Environmental Sensitivies of Quartz Oscillators," IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control, Vol. 39, Issue 2, pp. 241-249.

[7] M. Jurvansuu, J. Prokkola, M. Hanski and P. Perälä, "HSDPA Performance in Live Networks," IEEE International Conference on Communications, June 24-28 2007.

[8] J. Prokkola, M. Hanski, M. Jurvansuu and M. Immonen, "Measuring WCDMA and HSDPA Delay Characteristics with QoSMeT," IEEE International Conference on Communications, June 24-28 2007.

[9] Nethawk M5 analyzer, http://www.nethawk.fi, 10/2007.

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

[11] P. Perälä,"Multipoint Measurement for Bottleneck Localization on E2E Path", Master's thesis. University of Oulu, Department of electrical on Information Engineering, Finland, 2007.

[12] Converging Networks Laboratory, http://cnl.willab.fi, 10/2007.

[13] Distributed Internet Traffic Generator, http://www.grid.unina.it/software/ITG/, 10/2007.

[14] N. T. Kung and R. Morris, "Credit-based Flow Control for ATM Networks," IEEE Network, Vol. 9, Issue 2, 1995, pp. 40-48.

[15] H. Holma and A. Toskala, "HSDPA/HSUPA for UMTS: High Speed Radio Access for Mobile Communications", John Wiley & Sons, Ltd., West Sussex, England, 2006.

[16] Public Access Network Oulu, http://www.panoulu.fi, 10/2007.

[17] AirPcap 802.11 Wireless Packet Capture device by Cace Technologies. URL: http://www.cacetech.com/products/airpcap.htm, 10/2007.

[18] J. Jun, P. Peddabachagari and M.Sichitiu, "Theoretical maximum throughput of IEEE 802.11 and its applications," The Second IEEE International Symposium on Network Computing and Applications, April 16-18 2003, pp 249-256.

[19] ITU-T, "End-user multimedia QoS categories," ITU-T Recommendation G.1010.