Influence of the Packet Size on the One-Way Delay on the Down-link in 3G Networks

Patrik Arlos, Markus Fiedler

IEEE International Symposium on Wireless Pervasive Computing

2010 Mondena

Influence of the Packet Size on the One-Way Delay on the Down-link in 3G Networks

Patrik Arlos, Markus Fiedler

Blekinge Institute of Technology
37179 Karlskrona, Sweden
Patrik.Arlos@bth.se
Markus.Fiedler@bth.se

***Abstract*—The number of mobile broadband users is increasing. Furthermore, these users have high expectations into the capabilities of mobile broadband, comparable to those in fixed networks. On the other hand, the capacity assignment process on mobile access links is far from transparent to the user, and its properties need to be known in order to minimize the impact of the network on application performance. This paper investigates the impact of the packet size on the characteristics of the one-way delay for the down-link in third-generation mobile networks. For interactive and real-time applications such as VoIP, one-way delays are of major importance for user perception; however, they are challenging to measure due to their sensitivity to clock synchronization. Therefore, the paper applies a robust and innovative method to assure the quality of these measurements. We focus on the down-link as this is still the link that carries the most traffic to the user, and the quality of it will have a significant impact on all IP-based services.**

Results from measurements from several Swedish mobile operators reveal the possibility to partly control one-way delay and its variability by choosing appropriate packet sizes. In particular, packet sizes leading to the use of WCDMA entail significant but hardly varying one-way delays. On the other hand, we also show that HDSPA networks can deliver large amounts of data at rather high speed, but the cost is a huge variability in the one-way delay.

I. Introduction

Increasingly many devices use mobile connectivity for the exchange of data. Users expect the emerging mobile broadband to perform in a similar way as its fixed counterpart, no matter to which extent the medium is shared. In the third generation of mobile communications, represented by WCDMA and HSDPA, the per-user capacity allocation depends amongst others on the radio conditions, the user density, the mobility pattern, the offered traffic, etc. It is, however, not communicated explicitly towards user and application that might need this information for yielding the best performance, given the specific allocation. This imposes the need for end-to-end measurements with the goal to highlight network impact on the performance parameters of interest.

Given this background, this paper investigates the impact of the packet size on the one-way delay (OWD) for the down-link in third-generation mobile networks. In particular, the minimal OWD provides information about the best-possible performance with given settings, undisturbed by congestion, radio problems, etc. This is then correlated with statistics, such as mean and standard deviation. As OWD measurements are sensitive to clock synchronization, the paper presents a robust method to ensure exact OWD measurement across the mobile network.

In [1] the authors analyse the round trip time and OWD for operational 3G networks in Austria. While that study is similar in the way that the OWD via the down-link is studied, the methods used are different. The authors of [1] perform active testing and measurements on the same devices, while our measurements are entirely passive and will not affect any part of the network. Furthermore, they indicate a time stamp accuracy [2] of 0.5 ms, but this results in an accuracy of 1 ms for the OWD, while our methods accuracy is $\approx$ 100 ns, for the OWD values.

The paper is organized as follows. First, we describe the measurement method in Section II. Section III describes and discusses the experimental setup and analysis procedure. In Section IV, the down-link OWD in several Swedish networks are evaluated as a function of the packet size. Section V concludes the paper and points out future work.

II. Method

The fundamental problem when evaluating the one-way delay (OWD) is how to handle the clock synchronization. The OWD is, as such, simple to calculate. The OWD of the i^{th} packet, d_i, is calculated as:

$$d_i = T_{i,b} - T_{i,a} \tag{1}$$

where $T_{i,a}$ is the arrival time of the i^{th} packet at location a; correspondingly $T_{i,b}$ is the arrival time of the same packet at location b. In the general case, the time stamps $T_{i,x}$ are obtained from two different clocks. To get an unbiased OWD estimate, these clocks should be synchronized. In [3] the authors investigate the three main synchronisation methods NTP, GPS and IEEE1588 used for OWD measurements. Usually the Network Time Protocol (NTP) [4] is used. This enables the clocks to be synchronized within $10 - 20$ ms for WAN, and < 1 ms for LAN. If the synchronisation needs to be better, then a GPS solution is needed. Together with NTP, this allows a synchronisation in the order of 1 μs. The current state ot the art is to use Endace [5] DAG cards together with a GPS, then the theoretical synchronisation is in the range

978-1-4244-6857-7/10/$26.00 ©2010 IEEE

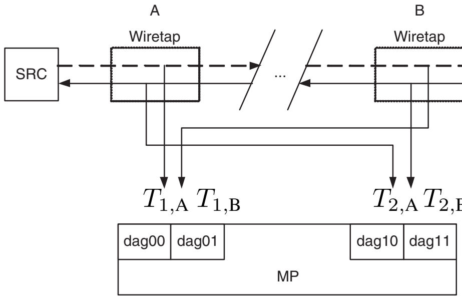

Fig. 1. Wiring method

of 60 ns. However, according to our own experience, this is difficult to obtain in practice, as we still have two independent clocks. In [6] the author described the internal functioning of the time-keeping in side of the DAG cards, and in [7] the authors describe a method to synchronize clocks across the Internet. Regardless of what method or technique used for synchronization, the OWD estimations can at the worst be twice that of the synchronization level [8].

Our method uses wiretaps and a special wiring in conjunction with DAG cards to obtain the time stamps from the same clock. In Figure 1 a schematic of the wiring is shown. When a packet is sent from SRC to DST it will travel across the upper wire (dashed line). As is passes the first wiretap A, a copy of the packet is made and is sent to the interface dag00, where it arrives it at time $T_{1,\mathrm{A}}$. At the same time the original packet makes its way across the network and eventually reaches wiretap B. Here, a copy is sent to interface dag01, where it is received at $T_{1,\mathrm{B}}$. Similarly, if a packet is sent from DST to SRC the packets are duplicated by the wiretaps and made available to the dag1x interfaces. The main drawback with this wiring is that we require close proximity between SRC and DST. The actual distance is determined by the technology that carries the traffic from the wiretap to the DAG cards.

Let t_0 be the time when the packet actually passes wiretap A, and t_1 when it passes wiretap B. Then $T_{1,\mathrm{A}} = t_0 + L_a/P_s$, where L_a is the cable length between wiretap A and dag0, and P_s is the propagation speed in that cable. Similarly, we define $T_{1,\mathrm{B}} = t_1 + L_b/P_s = t_0 + L/P_s + L_b/P_s$, where L_b is the cable distance from wiretap B to dag0, and L is the cable distance between wiretap A and B. The OWD is then obtained as:$\Delta = T_{1,\mathrm{B}} - T_{1,\mathrm{A}} = L/P_s + \frac{L_b - L_a}{P_s}$. So if we select $L_b = L_a$ we cancel the second factor and obtain the desired OWD between the wiretaps.

The method was applied and verified in [9], where we showed that the method could detect change in the transmission time in the range of nano seconds, hence the method is more than accurate enough for the measurements that were conducted. But the biggest advantage with the method is the perfect synchronization between sender and receiver side packets.

III. Setup

To evaluate the mobile networks, we used the setup shown in Figure 2. Here SRC is sending traffic to DST. This is

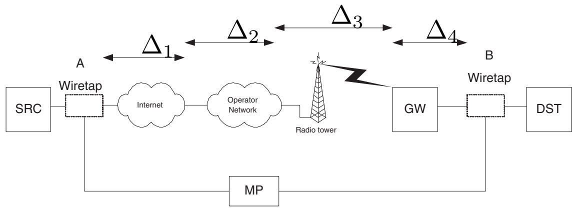

Fig. 2. Setup used in experiments.

done via the Gateway (GW) that uses a Huawei E220 USB modem to connect to the mobile network. In-between the SRC and Internet, we placed wiretap A. The other wiretap B is placed just in front of DST. The SRC and DST (both are P2-400 MHz with Linux 2.4 kernels) are connected with 10 Mbps full-duplex Ethernet cards (3Com). The GW is a Dual AMD Athlon 64 with 2 Gbytes of RAM (Windows XP SP2). The GW was configured for Internet sharing of the mobile network and no firewall was active, but we added port forwarding so that the traffic is forwarded to DST. The DST computer connected directly to the built-in Ethernet card (Broadcom) of the GW. The wiretaps feed into a Distributed Passive Measurement Infrastructure [10] enabled Measurement Point (MP) that stored the packet trace to file. Furthermore, the DAG cards were synchronised using both NTP and GPS.

A. *Traffic Generation*

To generate data we used a C++ program that sends UDP datagrams and allow us to control packet sending rate and datagram size. Furthermore, the program uses an application layer header with three fields. These fields allow us to separate experiments (experiment id), experiment run (run id), as well as packets within a particular experiment run (sequence number). The sequence number starts at zero and is incremented by one for each transmitted datagram. Based on these three fields, we can uniquely identify each packet, thus avoiding any ambiguities associated with hashing.

During the evaluation of the mobile networks we used two streams running in parallel. The first sends one packet of size K bytes every T_s second, which is done 200 times. It then waits for a fixed amount of time, and then starts to send another batch of 200 packets, this time with a larger packet size. The procedure is then repeated for all the packet sizes we wish to investigate. The second stream runs continously throughout the evaluation of all the different packet sizes, sending one 48 byte packet every 10 second. The purpose with this stream is to detect any time-of-day based variations in the network. As the purpose is to find the OWD, we do not want to stress the system so that it needs to queue our traffic. Using the two streams we will at most inject $1468 + 48$ bytes during one second.

B. *Delay Calculation*

As we are in control of both sender and receiver, we can easily identify both sending and receiving IP address as well as UDP port numbers. We then use the application header for the individual packet identification. Once identified, we can calculate the OWD for the individual packets. Using the same

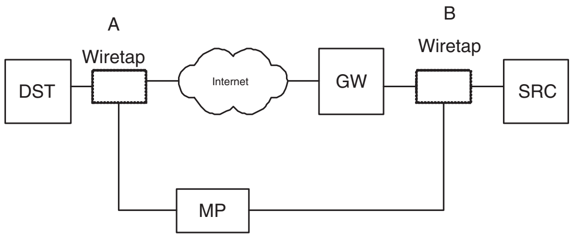

Fig. 3. Setup used when evaluating the GW.

notation as before, the delay would be calculated as defined in Equation 1. However, due to numerical issues [8], this is not recommended. It is better to use the following equation:

$$d_i = \widetilde{T_{i,a}} - \widetilde{T_{i,b}} \qquad \widetilde{T_{i,x}} = T_{i,x} - \lfloor T_1 \rfloor, \tag{2}$$

where $T_{i,x}$ is the arrival time of the i^{th} packet at location x; and T_1 is the arrival time of the first packet caught in this particular experiment. This will avoid having the time stamps truncated by the precision of the analysis tool.

C. Delay Components

The OWD that we will calculate has four contributors, see Figure 2. Δ_1 is the delay contribution by the Internet, Δ_2 that of the core network of the operator, Δ_3 that of the radio network, and the last contribution Δ_4 comes from the GW. Out of these four, we cannot measure or estimate Δ_2 and Δ_3 alone, as this means entering the domain of the operator.

We can estimate Δ_1 by using ping to the operator's Internet exchange. From our vantage point on the Internet, the operators are between five or six hops away, and between us and them we have the Swedish University Network (SUNET) with optical multi-gigabit links. Hence the impact of this will not be negligble, but it will be quite stable, the average RTT between DST and the operator Internet exchange is 15 ms for all three operators. Hence, as the links are symetrical, the OWD contribution will be around 7.5 ms. Furthermore, we can ignore the packet size as the links have such high capacity that the serialisation delay is neglible [11].

In order to quantify Δ_4, we designed a special experiment, shown in Figure 3. Instead of using the E220, we replaced it with a D-link DUB-E100 FastEthernet USB adaptor that allows us to connect directly to the destination through Ethernet. As we are using a USB NIC, the packets travelling across this NIC will receive the same treatment as those that are sent across the modem.

Figure 4 (a) show the CDF for Δ_4, while (b) shows the corresponding Δ_4 as a function of time. From the CDF we see that 99.99% of the packets see less than 0.4 ms of OWD when passing the GW. But there is one packet, as seen in (b), that experience a delay of 3.6 ms. In [9] we evaluated the up-link behaviour of the GW, and in that direction the GW exhibited a OWD with much higher variability, hence the direction of the stream does have some impact.

In Figure 5 we show the OWD through the GW for different packet sizes. We see that the OWD increases linearly with the packet size, as anticipated [11]. Furthermore, the largest

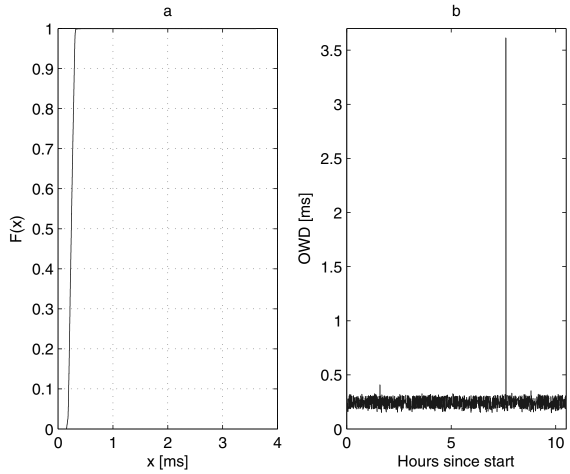

Fig. 4. CDF for Δ_4 through the GW(a), and the corresponding time trace (b).

minimum OWD is just above 1.3 ms. If we use the minimum delay as a base, we can construct a linear fit Δ_1 given in ms:

$$\Delta_1(L) = 8.303\text{e}{-4} \cdot L + 0.154 \tag{3}$$

Here L represents the IP packet length in bytes. The 0.154 ms represents the minimal time through the GW, and the constant (8.303e-4) corresponds roughly to the capacity of the interface, i.e. 10 Mbps. Similarly, one could construct an equation for the mean OWD. However, as we see the minimum and mean are quite close, and the major difference would be a slightly larger constant. Looking on the maximum values, we see that generally the maximum OWD tracks the minimum and mean values quite well, but for some packet sizes (i.e $204, 348, 412, \cdots 1068$ bytes), the maximum value increases, and in some cases it becomes almost 10 times larger than the mean value. As we have included the standard deviation in the graph, we see that these peaks are not that frequent. We have also investigated the individual traces, and from these it becomes obvious that it is one packet that exhibits this additional delay. If this would have been omitted the maximum value would also show a similar shape as the mean and minimum OWD. To see if the GW exhibited any periodic behaviour, we generated a graph that shows the OWD as a function of time. In this graph, shown in Figure 6, we have removed the impact that the packet size has on the OWD, using the model found in Equation 3. From it, it is clear that the GW does not exhibit any periodic behaviour that has a significant impact on the OWD. At maximum the additional OWD generated by the GW will be 4 ms, but in general much smaller.

As both Δ_1 and Δ_4 turned out to be significantly smaller than the OWD measured in the subsequent experiments, they are neglected from now on.

IV. Evaluation of Mobile Networks

We conducted experiments on three different Swedish operators. Two of them (A and B) share the radio access (RA), while the third (C) uses a different RA. The experiments focus

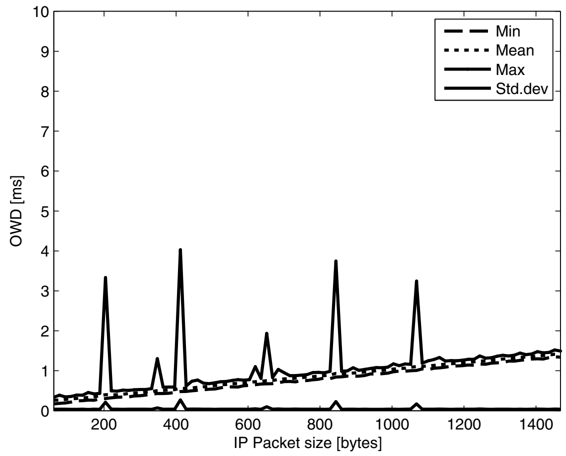

Fig. 5. OWD through the GW for different packet sizes.

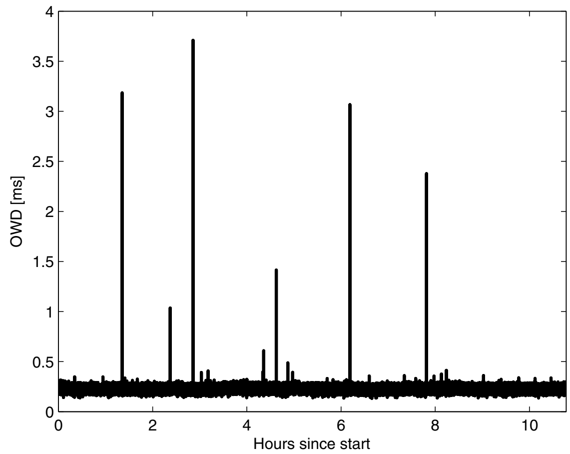

Fig. 6. OWD through the GW as a function of time.

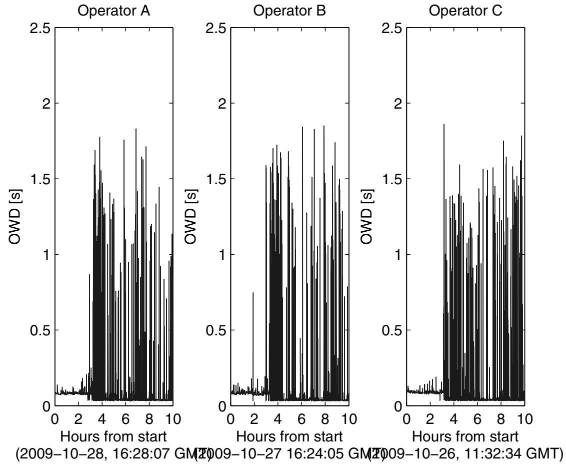

Fig. 7. Long term evaluation of the operators.

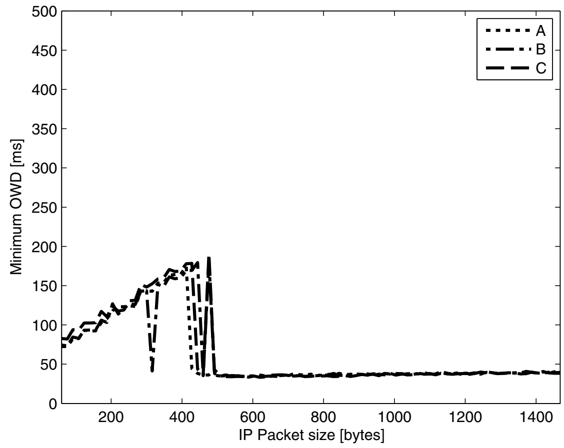

Fig. 8. Minimum one-way delay across different operators.

on the OWD of the down-link; furthermore, all experiments were done while the sender was stationary.

In Figure 7 we show the long term evaluation for all three operators. Generally all three seem to behave very similar, and we cannot see any indications of periodic patterns. Secondly, the largest OWD are below 2 seconds. We can also see two regions of different behaviour. In the first the OWD is relatively small, and with little variability, and in the second the OWD is even smaller, but the variability is significant. A possible reason is that after (around) three hours the first stream has reached a packet size that is so large that the mobile network changes from WCDMA into HDSPA mode, hence the decrease of the OWD, but also the increase in variability.

Figure 8 shows the minimum OWD as perceived by the different packet size. Here we see three regions, one that ranges from 60 to 400 bytes, then there is a transitional region up to 492 bytes, and the last one covers the remaining packet sizes. In the first region, the OWD increases almost linearly with the packet size, from approximately 75 ms up to 170 ms. Up to this packet size, all operators behaved rather similar, with small deviations on their OWD values. But, in the transitional phase, they behave different. Here, operator A drops from 174.5 ms for packet size 412 bytes to 44.4 ms for 428 bytes. It then continues to decrease until it reaches a value around 35 ms for packet sizes greater than 508 bytes. Operator B, on the other hand, goes from 180 ms at 444 bytes to 35 ms at 460 bytes, then back up to 181 ms for 476 bytes, and then down to 35 ms at 492 bytes. Operator C exhibits a similar behaviour, but it drops down to 44 ms already at 444 bytes, and then goes up to 190 ms at 476 bytes. This indicates that the network changes from a WCDMA mode to a HSDPA mode when the packets are in the transitional range, the actual point of change is not the same from operator to operator. Furthermore, operator B might have made temporary transistion already at 316 bytes, as here the OWD is 41 ms.

Now, we turn our attention to the mean OWD, shown in Figure 9. Again, we see two regions, one up to just above 400 bytes, where all three operators more or less exhibit the same OWD. Then in the second region, the mean OWD is varying significant, and the operators have quite different behaviors compared to each other. This is emphasised in Figure 10, where we show the standard deviation of the OWD. The regions are clear, in the first region the variability is small, but in the second region the standard deviation is big, and very often much bigger than the mean OWD. We observe packet sizes where the OWD standard deviation is around 700 ms,

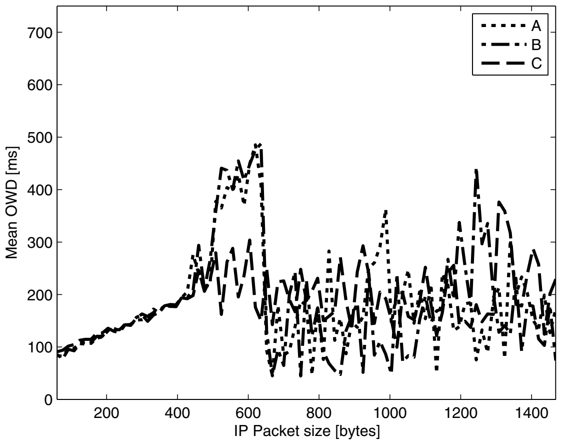

Fig. 9. Mean one-way delay across different operators.

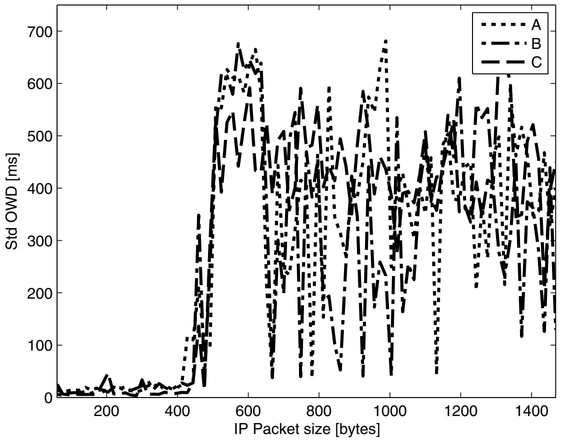

Fig. 10. Standard deviation of the one-way delay across different operators.

while the mean delay is 362 ms. This is emphasised even more in Figure 11, where the maximum OWD is displayed. There seems to be a level at 2100 ms for all operators, but on occation operator A and B need more than 3 seconds to deliver a packet. And given the small minimal OWD in combination with the large standard deviation, these long delays are not that frequent, as one would hope.

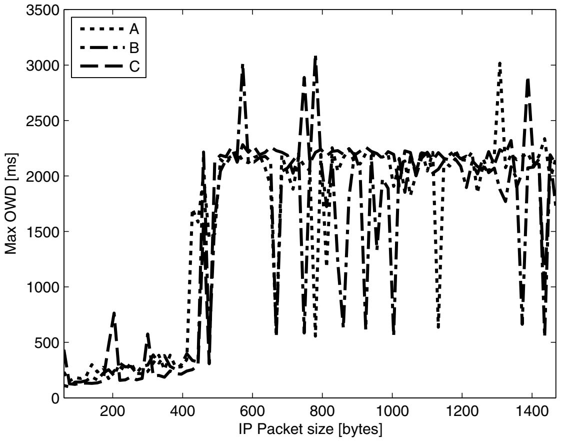

Fig. 11. Maximal one-way delay across different operators.

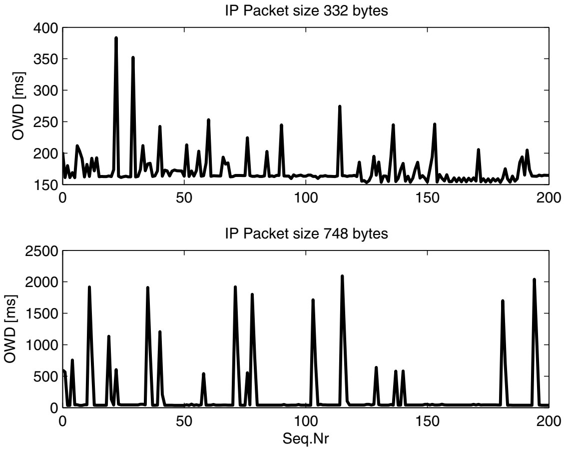

Fig. 12. A comparison between OWD for two different packet sizes, for the same operator (A).

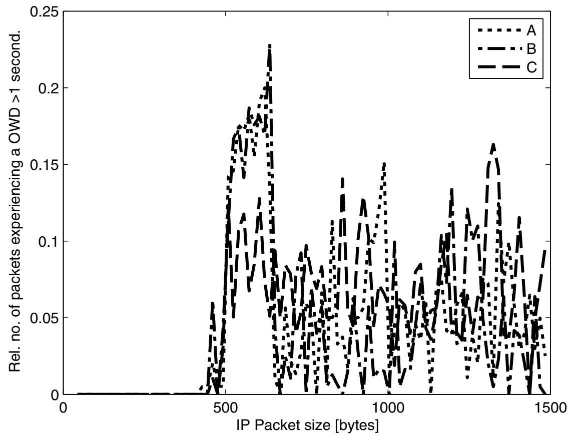

Fig. 13. The relative number of packets experiencing a OWD greater than 1 second.

To investigate the characteristics of the regions, we analysed two different packet sizes 332 bytes and 748 bytes. The corresponding OWDs are shown in Figure 12, the top graph shows 332 bytes/WCDMA mode, and the bottom shows 748 bytes/HDSPA mode. In WDCMA mode, the OWD does not go below 160 ms, and on two occations it becomes greater than 300 ms, but mostly stays below 200 ms, the mean is 172 ms and the standard deviation is around 28 ms. For the HSDPA mode this is not the case, here the minimum value is 36 ms, the mean is 188 ms, the maximum is 2.1 s, and it has a standard deviation of 415 ms.

In Figure 13, we show the relative number of packets that experience a OWD greater than one second. It shows us that for packet sizes greater than 400 bytes, up to 25% can experience this, and for the large majority of the packet sizes, this behaviour is observed for 5% of the packets.

The data shown in Figure 13, suggests that TCP should do better via WCDMA than via HSDPA, as the huge variations shown do not work in favour of a high throughput. But, our experience is that TCP downloads via HDSPA work rather well. To see if our low rate stream (one packet per second), receives a different treatment (cf. a TCP download),

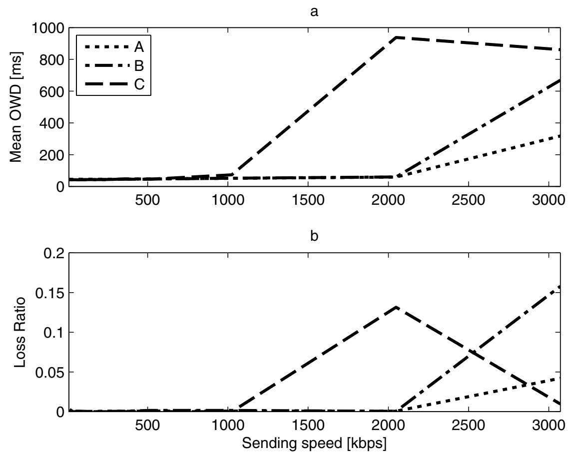

Fig. 14. The mean one-way delay (a) and loss ratio (b) as a function of sending speed.

we conducted one more experiment, where we fixed the packet size to 750 bytes, and then varied the sending rate 8, 16, 32, 48, 128, 256, 360, 512, 1024, 2048 and 3072 kbps (still using UDP as transport protocol). The results are shown in Figure 14, (a) shows the mean one-way delay, and (b) shows the loss ratios (number of lost packets/number of sent packets). From (a) we see that up to 1024 kbps, the mean OWD is quite small, less than 75 ms. Then the stream via operator C experiences 13% packet loss at 2048 kps, increasing the mean OWD to 937 ms, while A and B remain at a 60 ms and less than 0.5% packet loss. At the highest transmission speed, 3072 kbps, operator A and B increase both their loss and OWD values, while C lowers the loss ratio significantly but remains at a high value of 860 ms for the OWD.

V. Conclusions and Outlook

Based on quality-assured measurements from three Swedish mobile operators, this paper provides insights into the relationship between packet sizes and one-way delays, revealing the corresponding operator's resource allocation policy on the WCDMA/HSDPA down-link. Hereby the quality of the one-way delay (OWD) measurements has been assured by a specific set-up and cabling scheme between Endace DAG cards that avoids common clock synchronisation problems, and by a quantification of the delay contribution of the gateway feeding the mobile link.

We found that for all operators, the packet size has a pronounced influence on the OWD. Furthermore, we found that all three operators exhibit a clear and similar one-way delay pattern, one for WCDMA and another one for HSDPA. In WCDMA the packet size causes the OWD to grow linearly, here we also note that the standard deviation and extreme values are quite close to the mean OWD (with a few exceptions for the maximum value). In HSDPA the minimum OWD becomes quite small for all operators, less than 50 ms. But the variability of the delay grows extremely much, and in fact, for most packet sizes, the standard deviation is usually greater than the mean OWD. Furthermore, the typical maximum value reaches 2.1 s and some times the maximum is even higher. We also showed that the impact of the sending speed on the OWD was negligible as long as the operator supported speed was not exceeded. These results are similar to those presented in [1], but the differ in some important ways. In our case, the operators OWD are in the range of 35 ms to 3 s, the WCDMA region has a much sharper increase than that presented in [1], and in the HSDPA region our mean OWD seems to level off, while in [1] the OWD shows a slight increase. On the other hand, their OWD is in the range of 20–30 ms, while ours is between 35 and 500 ms. We also see a sharper lower boundary value, which is due to the higher accuracy of our method.

Future work will address a more detailed statistical analysis of the OWD as function of the packet size as well as investigations of the impacts of both parameters on TCP throughput.

Acknowledgements

We would like to thank Ravichandra Kommalapati for conducting the experiments.

References

[1] J. Fabini, W. Karner, L. Wallentin, and T. Baumgartner, "The illusion of being deterministic — application-level considerations on delay in 3G HSPA networks," in *NETWORKING '09: Proceedings of the 8th International IFIP-TC 6 Networking Conference*. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 301–312.

[2] P. Arlos and M. Fiedler, "A method to estimate the timestamp accuracy of measurement hardware and software tools," in *PAM2007*, ser. Lecture Notes in Computer Science, S. Uhlig, K. Papagiannaki, and O. Bonaventure, Eds., vol. 4427. Springer, 2007, pp. 197–206.

[3] L. Tomaciello, L. De Vito, and S. Rapuano, "One-way delay measurement: State of art," in *Instrumentation and Measurement Technology Conference, 2006. IMTC 2006. Proceedings of the IEEE*, April 2006, pp. 218–223.

[4] D. Mills, "Network time protocol (version 3) specification, implementation," United States, 1992.

[5] Homepage. Endace.com. [Online]. Available: http://www.endace.com

[6] S. Donnelly, "High precision timeing in passive measurements of data networks," Ph.D. dissertation, The University of Waikato, 2002.

[7] D. Veitch, S. Babu, and A. Pàsztor, "Robust synchronization of software clocks across the internet," in *IMC '04: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*. New York, NY, USA: ACM, 2004, pp. 219–232.

[8] P. Arlos, "On the quality of computer network measurements," Ph.D. dissertation, Blekinge Institute of Technology, 2005.

[9] P. Arlos and M. Fiedler, "Influence of the packet size on the one-way delay on the uplink in 3G networks," in *PAM2010*, ser. Lecture Notes in Computer Science, A. Krishnamurthy and B. Plattner, Eds., vol. 6932. Springer, 2010, pp. 61–70.

[10] P. Arlos, M. Fiedler, and A. A. Nilsson, "A distributed passive measurement infrastructure," in *PAM2005*, ser. Lecture Notes in Computer Science, C. Dovrolis, Ed., vol. 3431. Springer, 2005, pp. 215–227.

[11] D. Constantinescu, P. Carlsson, and A. Popescu, "One-way transit time measurements," Blekinge Institute of Technology, Research Report, 2004.

[12] C. Dovrolis, Ed., *Passive and Active Network Measurement, 6th International Workshop, PAM 2005, Boston, MA, USA, March 31 - April 1, 2005, Proceedings*, ser. Lecture Notes in Computer Science, vol. 3431. Springer, 2005.

[13] A. Krishnamurthy and B. Plattner, Eds., *Passive and Active Network Measurement, 11th Internatinoal Conference, PAM 2010, Zurich, Switzerland, April 7-9, 2010, Proceedings*, ser. Lecture Notes in Computer Science, vol. 6932. Springer, 2010.

[14] S. Uhlig, K. Papagiannaki, and O. Bonaventure, Eds., *Passive and Active Network Measurement, 8th Internatinoal Conference, PAM 2007, Louvain-la-neuve, Belgium, April 5-6, 2007, Proceedings*, ser. Lecture Notes in Computer Science, vol. 4427. Springer, 2007.